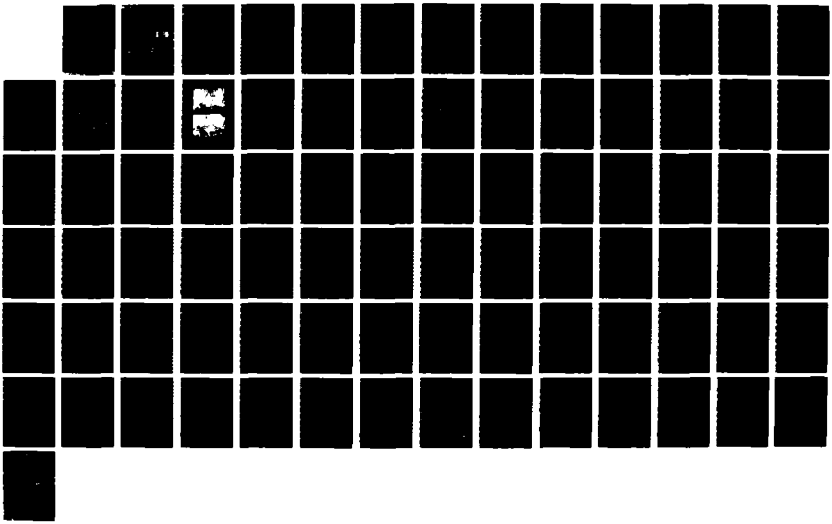
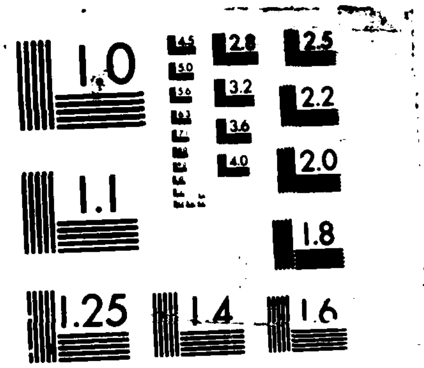
AD-A181 415 SUPERVISORY MANIPULATION FOR ASSEMBLING MECHANICAL PARTS WHILE COMPENSATI (U) MASSACHUSETTS INST OF TECH CAMBRIDGE DEPT OF MECHANICAL ENGIN H HIRABAYASHI 1/1

UNCLASSIFIED JUN 81 N00014-77-C-0256 F/G 23/2 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

196-152

(1)

SUPERVISORY MANIPULATION FOR ASSEMBLING MECHANICAL PARTS WHILE COMPENSATING FOR RELATIVE MOTION

HISAKI HIRABAYASHI

JUNE 1981

Hitachi, Ltd.
Japan

Grant 04-7-158-44079
MIT SEA GRANT PROGRAM
Office of Sea Grant
National Oceanographic and
Atmospheric Administration
Department of Commerce

Contract N00014-77-C-0256
Work Unit Number NR196-152
Engineering Psychology Programs
OFFICE OF NAVAL RESEARCH

Supervisory Manipulation for Assembling Mechanical
Parts While Compensating for Relative Motion

by

Hisaaki Hirabayashi

Kogakushi, Waseda University
(1972)

Kogakushushi, Waseda University
(1974)

SUBMITTED TO THE DEPARTMENT OF
MECHANICAL ENGINEERING
IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS OF THE
DEGREE OF

MASTER OF SCIENCE IN
MECHANICAL ENGINEERING

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 1981

Signature of Author Hisaaki Hirabayashi
Department of Mechanical Engineering
June 22, 1981

Certified by Thomas B. Sheridan
Thomas B. Sheridan
Thesis Supervisor

Accepted by W M Rohsenow
Warren M. Rohsenow
Chairman, Departmental Graduate Committee

Supervisory Manipulation for Assembling Mechanical Parts While Compensating for Relative Motion

by

Hisaaki Hirabayashi

Submitted to the Department of Mechanical Engineering on June 22, 1981 in partial fulfillment of the requirements for the Degree of Master of Science in Mechanical Engineering

ABSTRACT

This research is to develop and demonstrate the capability for a manipulator system to automatically compensate for random motion of the object being manipulated. This is done by means of a computer and a "measurement arm", a multi-degree-of-freedom position sensor independent of the manipulator itself.

Following preliminary experiments ~~of~~ Dr. K. ~~Tani~~ which presupposed perfect measurement, we developed the position sensor and the Jacobian matrices of approximation necessary to interject and transform the measurement to enable control. This report describes the interaction of the 6 degree-of-freedom sensor, and the Jacobian matrices of first order approximation. Evaluation tests were done for simple motions. As the result of the tests, we found the errors acceptable, and believe that this technique is useful for this type of compensation.

Thesis Supervisor: Dr. Thomas B. Sheridan
Title: Professor of Mechanical Engineering

ACKNOWLEDGMENTS

First I most gratefully acknowledge the instruction and guidance of Professor Thomas B. Sheridan. I am also thankful to Mr. Don Fyler and Mr. Ahmet Buharali for the kind assistance in using the computer, Mr. Dana Yoerger for his constructive comments, Dr. Anatoly Farberov for his aid to mechanism and the staff and the students of the Man-Machine Laboratory, and Dr. Kazuo Tani who performed the preliminary experiments which led to my thesis.

I would like to acknowledge Hitachi, Ltd. Japan, for supporting me financially and finally express my appreciation to my wife Tomoko for her encouragement and continual support during my study at M.I.T.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

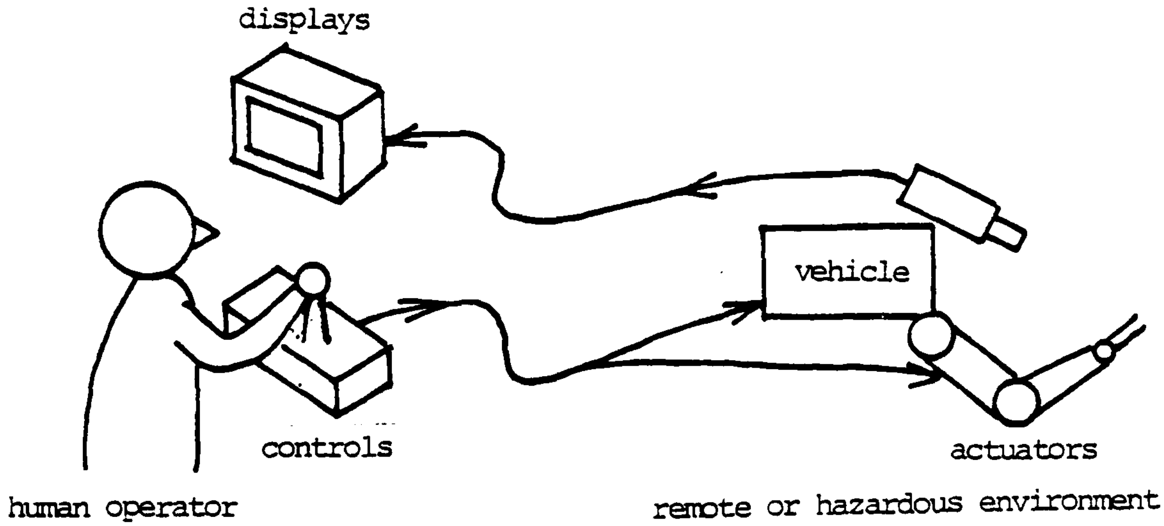
CHAPTER 1

INTRODUCTION TO SUPERVISORY MANIPULATION

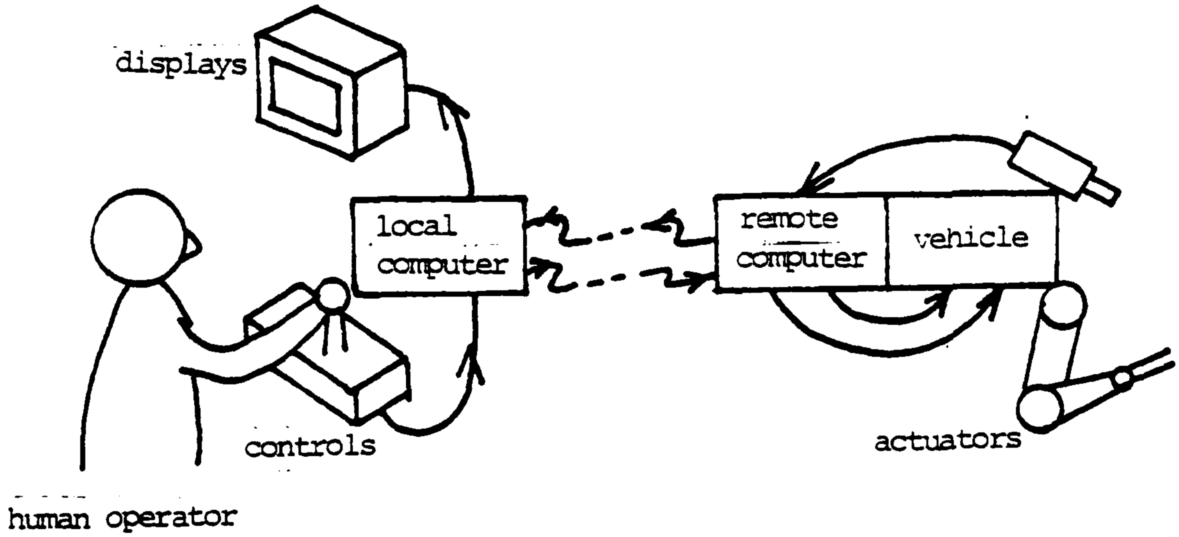
1.1 Supervisory Control

Recently the need has increased for operating vehicles and devices far from the surface of the earth such as underwater submersibles and space shuttles. These operations are so difficult that fully unmanned systems can not carry out them. Consequently, for these operations the man is required to be within the systems to make some decisions. These manned systems are divided into two categories: one is direct control and another is supervisory control. We can explain the difference as follows, by making use of Fig. 1.1.[1] In direct control, the human directly controls, over the communication link, the separate propulsive actuators of the vehicle, the actuators for the separate degrees of freedom of the manipulator and the actuators of the video camera. The video picture is sent back directly to the operator. The "hand control" can be a master-slave positioning replica or a rate joystick. In supervisory control a computer is added to the teleoperator, and for short periods and limited circumstances the teleoperator can function autonomously. Then the terminology, teleoperator, is defined as follows: A vehicle having sensors and actuators for mobility and/or manipulation, remotely controlled by a human operator, and thus enabling him to extend his sensory - motor function to remote or hazardous environments.

One example of direct control is the conventional master - slave arm system, where the operator has to make all decisions for control on the basis of all information from teleoperator in a short time. It will



a) Direct Control



b) Supervisory Control

Fig. 1.1 Direct and Supervisory Control of a Teleoperator

make him do too much work and might lead to misoperations in complicated systems. Alternatively, supervisory control may be advantageous to achieve faster or more accurate control, or to control simultaneously in more degrees - of - freedom than the operator can achieve by direct control, or relieve him of tedium. This is why a computer is included to carry out a part of the work in addition to the operator. Strictly speaking these computer roles are divided into four categories, as in Fig. 1.2 [1]:

1) it can extend his capabilities to help the teleoperator accomplish more than he alone were in control.
2) it can relieve him of some control tasks.
3) it can provide back-up by taking over control for a short time if feedback is lost.
4) it can replace him when a task is too dull.

We have explained supervisory control so far in comparison with the counterpart which is the manned direct control without computer. Next it is also important to explain supervisory control in comparison with the work of divers or manned submersibles in the sea.

About ten years ago divers seemed to have an advantage over manned work - vehicles with manipulators in terms of maneuverability, manipulation, tactible sensing, and covertness. Because of smaller unmanned vehicles and computers, only manipulation, sensing and cognition still remain the primary advantage for the divers.

As to the comparison between teleoperators and manned submersibles, because of the remarkable progress in television cameras and communication channels, the major difference remaining between manned submersibles and

Roles of Computer
(L-load or task, H-human, C-computer)

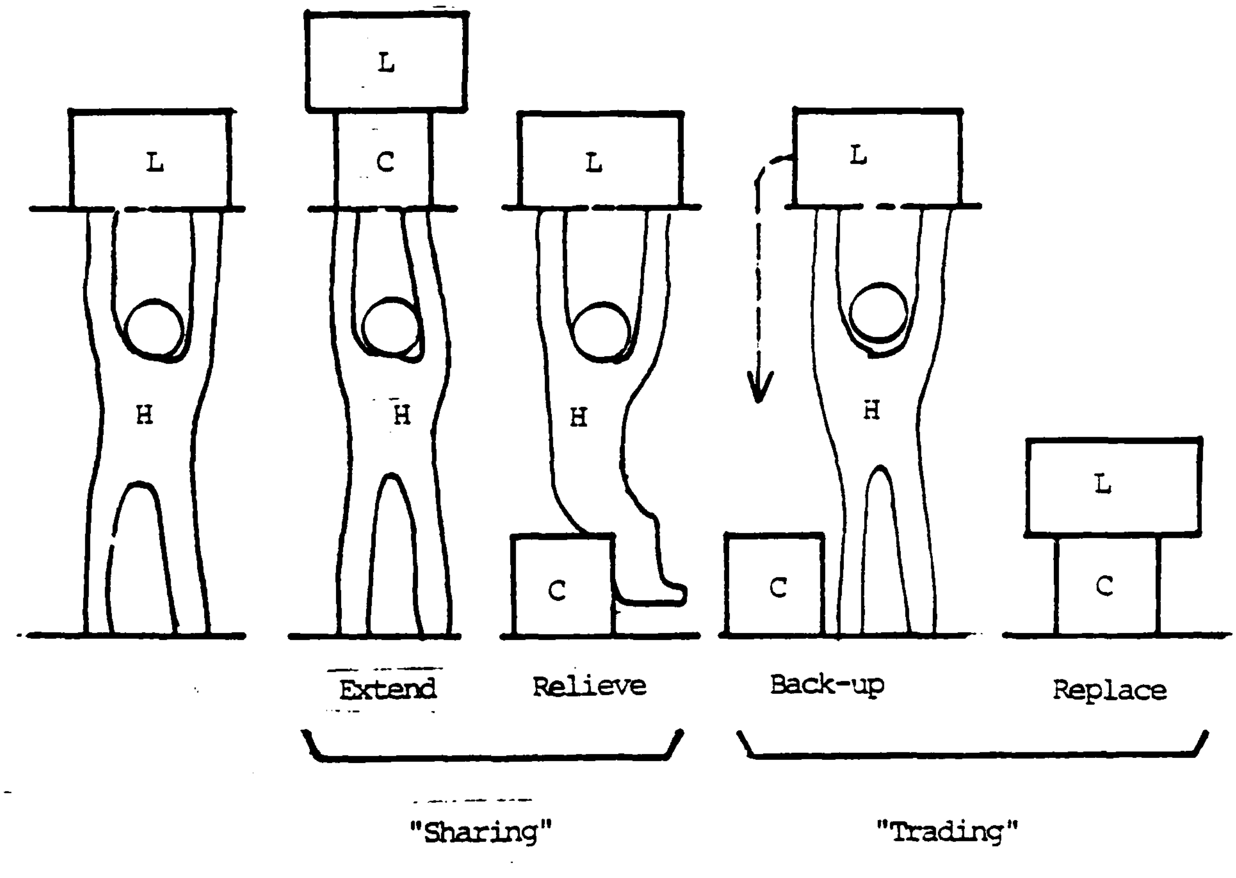


Fig. 1.2 Roles of Computer

teleoperators are cost and safety. The pressure vessel and life-support equipment make the manned submersible much more costly than the same vehicle without the pressure vessel and life-support equipment but with remote control instead.

The above-mentioned explanations show that supervisory control is getting to have an advantage in terms of technology and cost compared with divers and manned submersibles.

1.2 Compensation

There are many problems to be solved in the field of the supervisory control. As an example, in the project we concentrate on remote manipulation by the supervisory control, particularly on manipulation with automatic compensation, which belongs to the 1st category in computer roles said in 1.1, for moving targets.

As far as compensation is concerned, a first step has been achieved so far in our laboratory by Tani.[2] His work demonstrated that automatic manipulator compensation for relative motion between manipulated object and manipulator base made master - slave manipulation easier. For his work, he used a hardware system which consisted of master/slave manipulator, a moving table for the moving objects, and a computer controlling both the manipulator and the table. By means of the method of resolved motion rate control, his software system allowed the master/slave operation with object motion compensation under computer control. Tani's experiments were of three kinds: no object motion, compensation for the object motion, and no compensation. The comparison of the situations with the compensation without it showed that the compensation reduced the operation time or increased the accuracy in some tasks.

CHAPTER 2

PURPOSE OF THIS RESEARCH

2.1 Compensation with Position Measuring

The manipulator compensation done so far presupposed a perfect measurement of the relative motion between manipulated object and manipulator base. The purpose of this project is to extend the compensation by using an experimental 6 degree-of-freedom passive "measurement arm" having a simple gripper, but otherwise flaccid. The operator positions it with the actual manipulator. We explain it in Fig. 2.1.

$$\vec{A} = \vec{T} + \vec{R} \tag{2.1}$$

where

$\vec{A}$: Absolute Position of the Object with respect to the Manipulator Base

$\vec{T}$: Table Position with respect to the Manipulator Base

$\vec{R}$: Relative Position of the Object with respect to the Table

As shown in Fig. 2.1.a, in the case that the table where the object is mounted is fixed, all the operator has to do is to pick up and place the object with direct visual feedback. On the other hand, as shown in Fig. 2.1.b, in the case that the table moves, he is required to do more to achieve the same operation. That is, as we showed in Fig. 2.1.c he has to pick it up and to place it by taking account of the movement of both table and object. If there were some function which enabled him to operate as if the table were fixed, it would be very convenient for him.

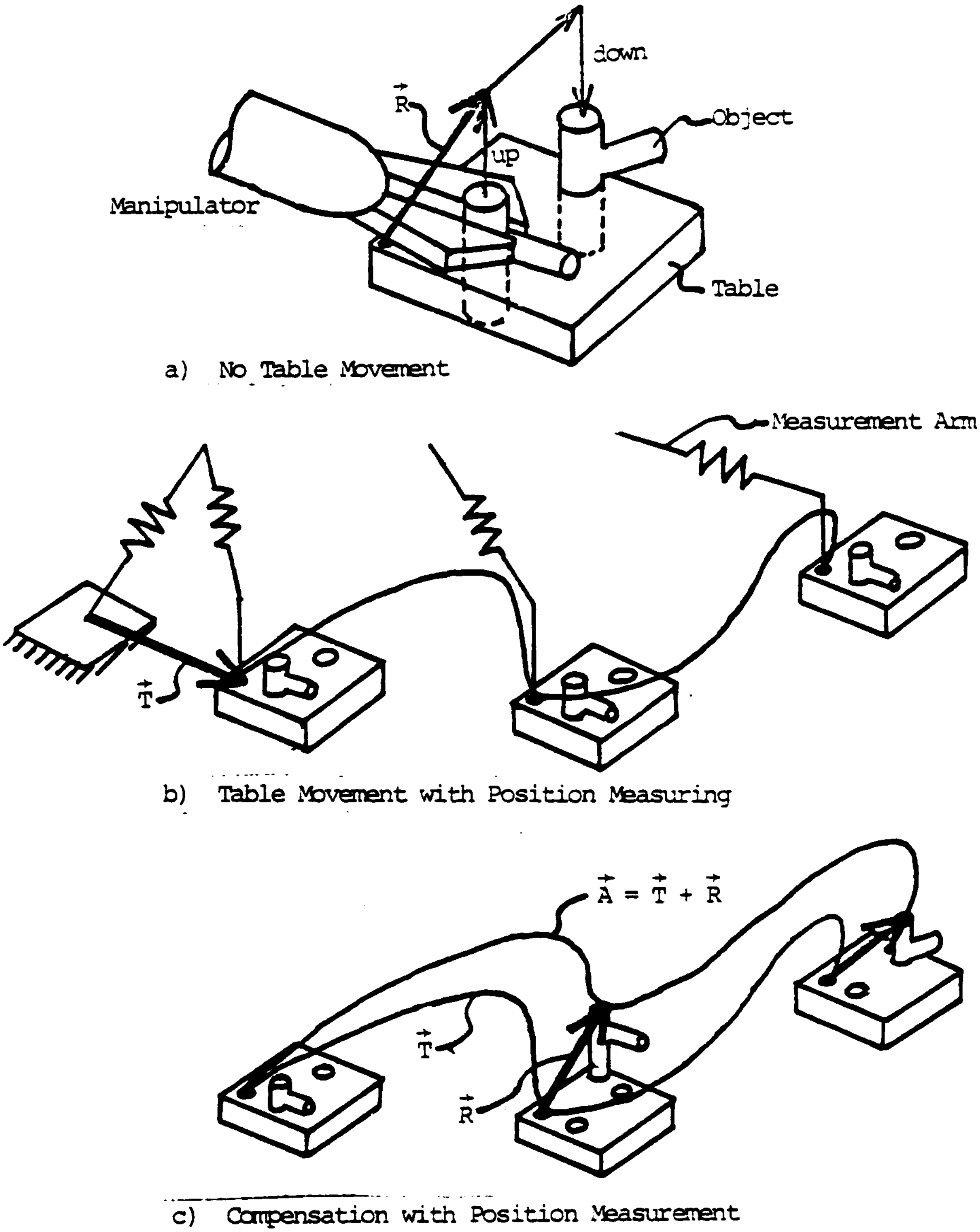


Fig. 2.1 Compensation with Position Measuring

In this project, I define compensation as this function. According to the above mentioned notation, the compensation means to change A from $\vec{T} + \vec{R}$ to $\vec{R}$. In other words it means subtraction of $\vec{T}$ from $\vec{A}$.

2.2 Function of Measurement Arm

When it comes to subtraction of $\vec{T}$ from $\vec{A}$, $\vec{T}$ has to be measured while $\vec{A}$ can be controlled by the operator. In order to measure $\vec{T}$, that is, table position with respect to the manipulator base, we introduced the measurement arm to the system. Fig. 2.2 by means of 6 potentiometers, are set in each joint of the arm, we can measure each joint angle; therefore, we can estimate the position and orientation of the table with the help of some geometrical calculations. With the aid of computation and by making use of the measurement arm, the operator can pick up and place the object on the moving table as if the table were fixed.

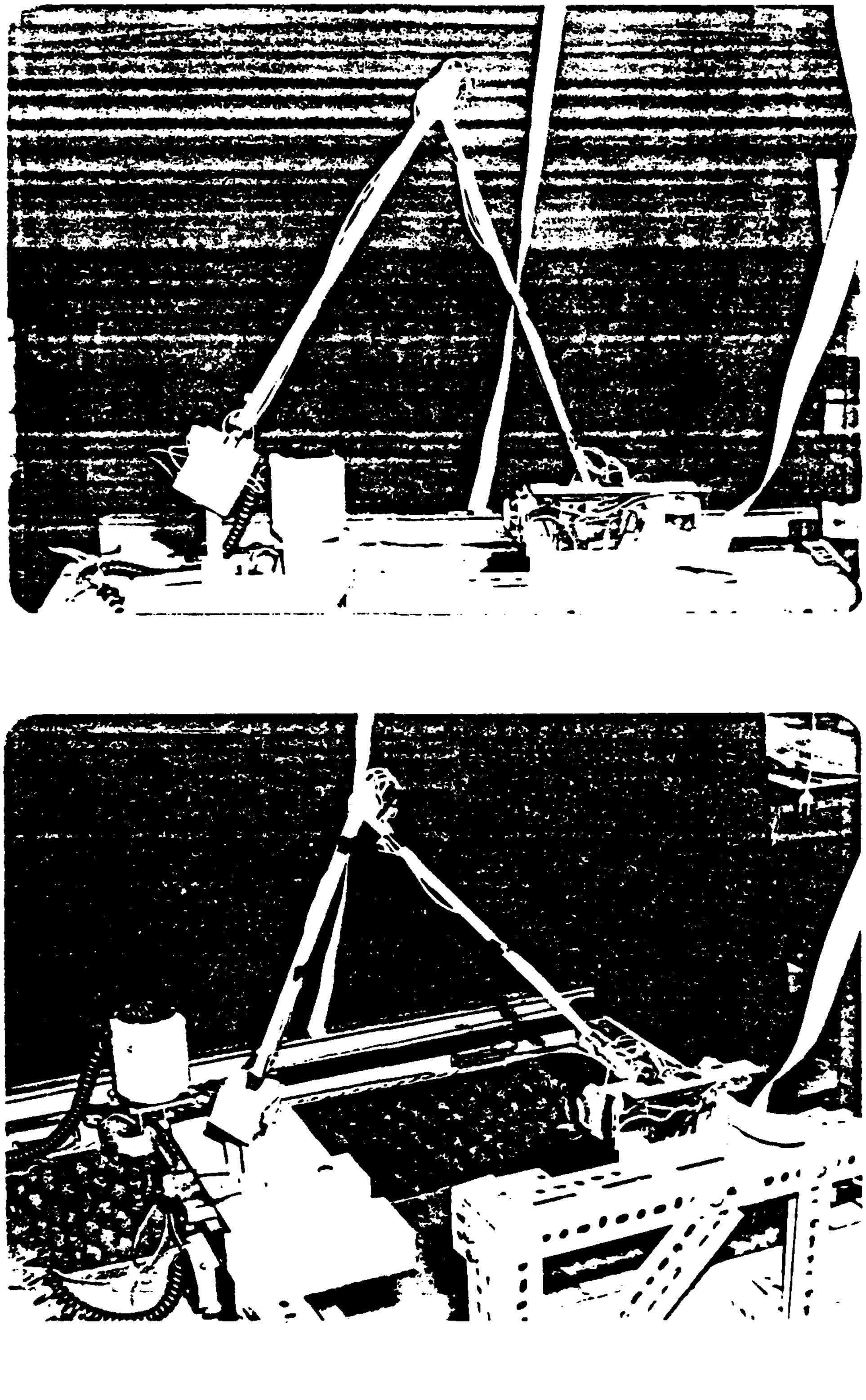


Fig. 2.2 Measurement Arm

CHAPTER 3

METHOD

3.1 Control Theory of Manipulation

At first, we explain the control of manipulation. It is the control of $\vec{A}$: Absolute Position of the Object with respect to the Manipulation Base, illustrated in Chapter 2. In Sec. 3.1 the general method is stated and in Sec. 3.2 the approximated method we used is stated.

Generally speaking, an unconstrained rigid body has six independent degree of freedom: three independent translation components and three independent rotation components.[3]

Since the hand of a manipulation is rigid body, it needs six independent components to be fixed in the space. We define the following six components: x, y and z which show the translation P of the hand and α, β and γ which show the orientation or rotation of the hand. In order to describe these six parameters, we use the coordinate system whose notation is given by T. Brooks.[4] Matrix ${}^{m}A_{n}$ means the transformation from the m th frame to the n th frame. For example the transformation from the hand frame (6 th) to the vehicle frame (0 th) is given as

$$ {}^{0}A_{6} = {}^{0}A_{1}\ {}^{1}A_{2}\ {}^{2}A_{3}\ {}^{3}A_{4}\ {}^{4}A_{5}\ {}^{5}A_{6} \tag{3.1} $$

Each frame is defined in Fig. 3.1 and components of each matrix are shown in Table A.1.

According to this notation, x, y and z are the vehicle coordinates at point P. With the definitions that the x_m axis means the x axis in the

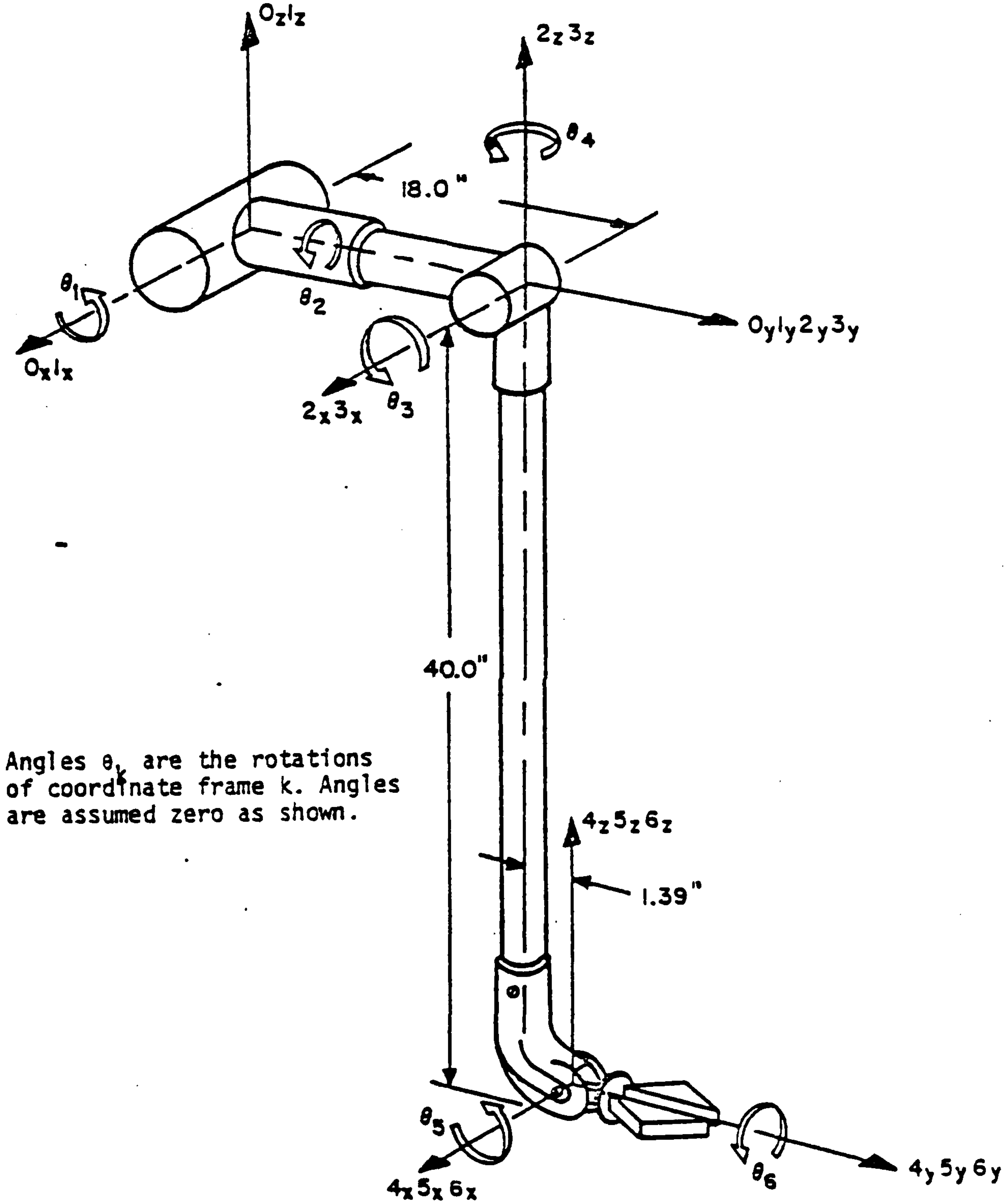


Fig. 3.1 Definitions of Coordinate Systems and Rotation Angles of the Manipulator (from Brooks [4])

m th frame and the $x_m y_m$ plane means the plane containing the x_m axis and the y_m axis, we define the rotation α, β and γ in Fig. 3.2. The rotation α is the angle between the $y_o z_o$ plane and the plane which contains the y_6 axis and which is perpendicular to the $x_o y_o$ plane. The rotation α, β and γ are defined in Fig. 3.2.

If all joints θ_p through θ_6 (θ_k specifies the rotation of the k th frame with the respect to the k - 1 the frame) are given, these six parameters are found as follows.

$$\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = {}^0A_6 \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = {}^0A_4 \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \tag{3.2}$$

$$\alpha = \tan^{-1} \left(\frac{x_{hy} - x}{y_{hy} - y}\right) \tag{3.3}$$

$$\beta = \tan^{-1} \left(\frac{z_{hy} - z}{\sqrt{(x_{hy} - x)^2 + (y_{hx} - y)^2}}\right) \tag{3.4}$$

$$\gamma = \tan^{-1} \left(\frac{z_{hx} - z}{\sqrt{(x_{hx} - x)^2 + (y_{hx} - y)^2}}\right) \tag{3.5}$$

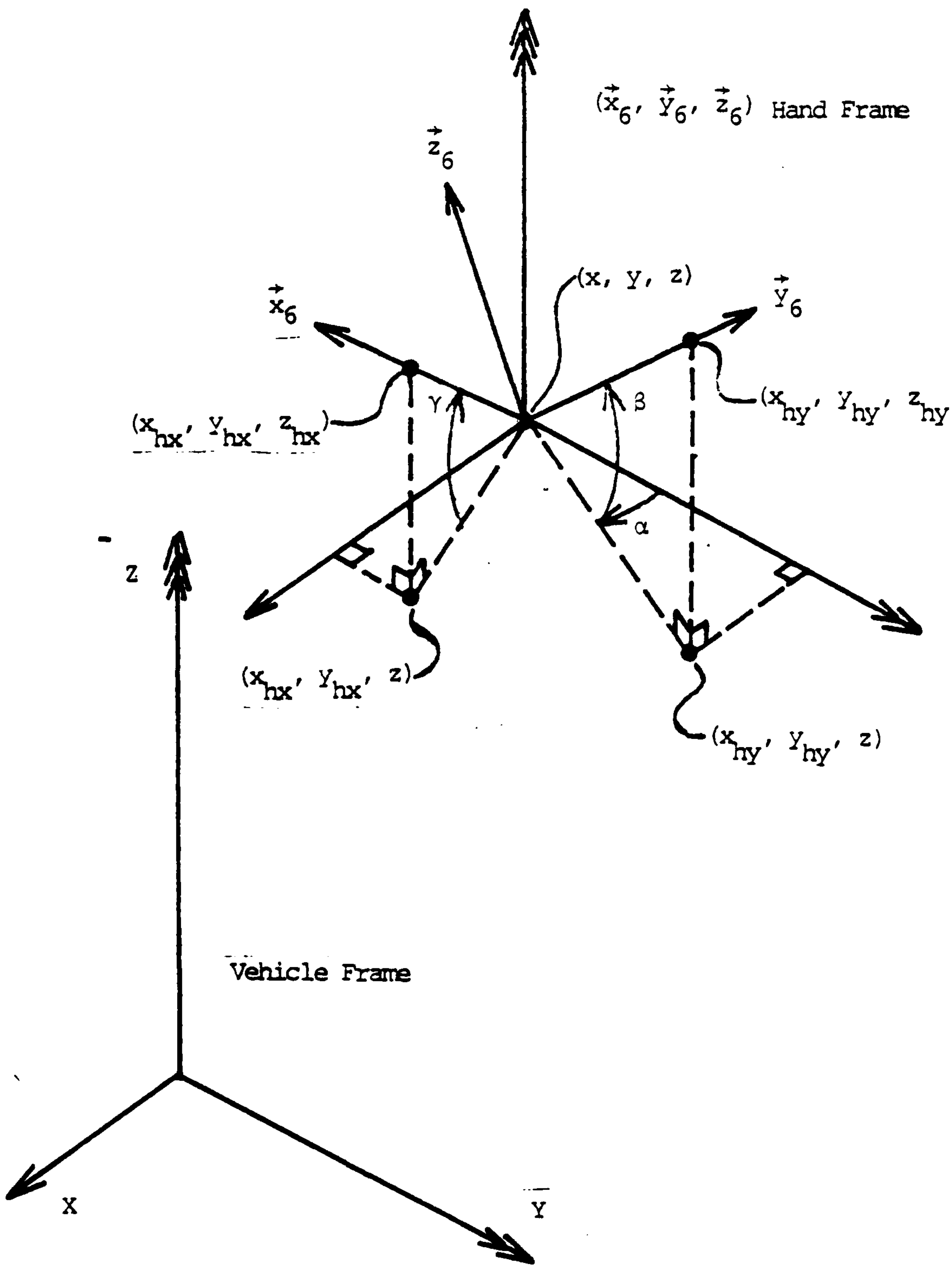


Fig. 3.2 Orientation of Hand

where

$$\begin{bmatrix} x_{hy} \\ y_{hy} \\ z_{hy} \\ 1 \end{bmatrix} = {}^0A_6 \begin{bmatrix} 0 \\ 1 \\ 0 \\ 1 \end{bmatrix} = \vec{y}_6 \tag{3.6}$$

$$\begin{bmatrix} x_{hx} \\ y_{hx} \\ z_{hx} \\ 1 \end{bmatrix} = {}^0A_6 \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \vec{X}_6 \tag{3.7}$$

In contrast with above procedure, it is difficult to find θ_1 through θ_6 when x, y, z, α, β and γ are given. Since we want to keep the orientation of the hand parallel to the $x_o y_o$ plane, we have to give

$$\beta = 0 \tag{3.8}$$

$$\gamma = 0 \tag{3.9}$$

From those equations, we find θ_5 and θ_6 as follows. From equations (3.4) and (3.5), $\beta = 0$ and $\gamma = 0$ mean $z_{hy} - z = 0$ and $z_{hx} - z = 0$, respectively. That is,

$$z_{hy} - z = (0,0,1,0)\ {}^0A_6 \begin{bmatrix}0\\1\\0\\1\end{bmatrix} - (0,0,1,0)\ {}^0A_6 \begin{bmatrix}0\\0\\0\\1\end{bmatrix}$$

$$= (0,0,1,0)\ {}^0A_6 \begin{bmatrix}0\\1\\0\\0\end{bmatrix} = a_{32} = 0$$

$$\therefore \theta_5 = \tan^{-1} [-(\sin\theta_1 \cos\theta_3 \cos\theta_4 + \cos\theta_1 \cos\theta_2 \sin\theta_3 \cos\theta_4 + \cos\theta_1 \sin\theta_2 \sin\theta_4)/(\cos\theta_1 \cos\theta_2 \cos\theta_3 - \sin\theta_1 \sin\theta_3)] \quad (3.10)$$

$$z_{hx} - z = (0,0,1,0)\ {}^0A_6 \begin{bmatrix}1\\0\\0\\1\end{bmatrix} - (0,0,1,0)\ {}^0A_6 \begin{bmatrix}0\\0\\0\\1\end{bmatrix}$$

$$= (0,0,1,0)\ {}^0A_6 \begin{bmatrix}1\\0\\0\\0\end{bmatrix} = a_{31} = 0$$

$$\therefore \theta_6 = \tan^{-1} \Big\{(-\cos\theta_1 \sin\theta_2 \cos\theta_4 + \sin\theta_1 \cos\theta_3 \sin\theta_4 + \cos\theta_1 \cos\theta_2 \sin\theta_3 \sin\theta_4)/[\cos\theta_1 \cos\theta_2 \cos\theta_3$$

$$- \sin\theta_1 \sin\theta_3) \cos\theta_5 - (\sin\theta_1 \cos\theta_3 \cos\theta_4 +$$

$$\cos\theta_1 \cos\theta_2 \sin\theta_3 \cos\theta_4 + \cos\theta_1 \sin\theta_2 \cos\theta_4)$$

$$\sin\theta_5]\Big\} \qquad (3.11)$$

Where a_{mn} represents an element in matrix ${}^0A_6$. On the other hand, from (3.3) α is expressed as

$$\tan\alpha = \frac{(1,0,0,0)\ {}^0A_6 \begin{bmatrix}0\\1\\0\\1\end{bmatrix} - (1,0,0,0)\ {}^0A_6 \begin{bmatrix}0\\0\\0\\1\end{bmatrix}}{(0,1,0,0)\ {}^0A_6 \begin{bmatrix}0\\1\\0\\1\end{bmatrix} - (0,1,0,0)\ {}^0A_6 \begin{bmatrix}0\\0\\0\\1\end{bmatrix}}$$

$$= \frac{a_{12}}{a_{22}} = [\sin\theta_2 \sin\theta_3 \cos\theta_4 - \cos\theta_2 \sin\theta_4$$

$$+ \sin\theta_2 \cos\theta_3 \tan\theta_5]/[\cos\theta_1 \cos\theta_3 \cos\theta_4$$

$$- \sin\theta_1 \sin\theta_2 \sin\theta_4 - \sin\theta_1 \cos\theta_2 \sin\theta_3 \cos\theta_4$$

$$-(\cos\theta_1 \sin\theta_3 + \sin\theta_1 \cos\theta_2 \cos\theta_3) \tan\theta_5] \qquad (3.12)$$

To eliminate θ_5, substitution of (3.10) to (3.12) gives

$$\tan\alpha = (-\cos\theta_1 \cos\theta_3 \sin\theta_4 - \sin\theta_1 \sin\theta_2 \cos\theta_4 + \sin\theta_1 \cos\theta_2 \sin\theta_3 \sin\theta_4)/(\cos\theta_2 \cos\theta_4 + \sin\theta_2 \sin\theta_3 \sin\theta_4) = S \tag{3.13}$$

By means of S instead of , we define P.

$$P = S(\cos\theta_2 \cos\theta_4 + \sin\theta_2 \sin\theta_3 \sin\theta_4) + \sin\theta_1 \sin\theta_2 \cos\theta_4 + \cos\theta_1 \cos\theta_3 \sin\theta_4 - \sin\theta_1 \cos\theta_2 \sin\theta_3 \sin\theta_4 \tag{3.14}$$

Now that we give four parameters: x, y, z and P, to find θ_1 through θ_4, the equations are rearranged as

$$x = 1.39\,(\sin\theta_2 \sin\theta_3 \cos\theta_4 - \cos\theta_2 \sin\theta_4) - 40 \sin\theta_2 \cos\theta_3 \tag{3.15}$$

$$y = 1.39\ (\cos\theta_1 \cos\theta_3 \cos\theta_4 - \sin\theta_1 \cos\theta_2 \sin\theta_3 \cos\theta_4$$

$$- \sin\theta_1 \sin\theta_2 \sin\theta_4) + 40\ (\sin\theta_1 \cos\theta_2 \cos\theta_3$$

$$+ \cos\theta_1 \sin\theta_3) + 18 \cos\theta_1 \qquad (3.16)$$

$$Z = 1.39\ (\sin\theta_1 \cos\theta_3 \cos\theta_4 + \cos\theta_1 \cos\theta_2 \sin\theta_3 \cos\theta_4$$

$$+ \cos\theta_1 \sin\theta_2 \sin\theta_4) + 40\ (-\cos\theta_1 \cos\theta_2 \cos\theta_3$$

$$+ \sin\theta_1 \sin\theta_3) + 18 \sin\theta_1 \qquad (3.17)$$

$$P = S(\cos\theta_2 \cos\theta_4 + \sin\theta_2 \sin\theta_3 \sin\theta_4)$$

$$+ \sin\theta_1 \sin\theta_2 \cos\theta_4 + \cos\theta_1 \cos\theta_3 \sin\theta_4$$

$$- \sin\theta_1 \cos\theta_2 \sin\theta_3 \sin\theta_4 \qquad (3.18)$$

It is quite difficult to solve equations (3.15)-(3.18), then we introduce the relation between the total differential and the partial differential of a function.[5]

The total differential of f is defined by the equation,[6]

$$df = \frac{\partial f}{\partial x} dx + \frac{\partial f}{\partial y} dy + \frac{\partial f}{\partial z} dz \qquad (3.19)$$

where f is a function of three variables x, y and z, and $\frac{\partial f}{\partial x}$ is the partial derivative with respect to x.

If x, y and z are all functions of a single variable, say t, then the dependent variable f may also be considered as truly a function of the one independent variable t.

Since only one independent variable is present, df/dt has a meaning and it can be shown, by appropriate limiting processes, that

$$\frac{df}{dt} = \frac{\partial f}{\partial x}\frac{dx}{dt} + \frac{\partial f}{\partial y}\frac{dy}{dt} + \frac{\partial f}{\partial z}\frac{dz}{dt} \quad . \tag{3.20}$$

For a short period of time Δt, it can be replaced by

$$\frac{\Delta f}{\Delta t} = \frac{\partial f}{\partial x}\frac{\Delta x}{\Delta t} + \frac{\partial f}{\partial y}\frac{\Delta y}{\partial t} + \frac{\partial f}{\Delta z}\frac{\Delta z}{\Delta t} \quad . \tag{3.21}$$

Consequently, we get

$$\Delta f = f_x \, \Delta x + f_y \, \Delta y + f_z \, \Delta z \tag{3.22}$$

where $f_x = \partial f/\partial x$.

Applying the relation to our problem gives

$$\underbrace{\begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \\ \Delta p \end{bmatrix}}_{\Delta \vec{x}} = \underbrace{\begin{bmatrix} x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & & \vdots \\ z_1 & & & \vdots \\ p_1 & - & - & p_4 \end{bmatrix}}_{J(\vec{\theta})} \cdot \underbrace{\begin{bmatrix} \Delta\theta_1 \\ \Delta\theta_2 \\ \Delta\theta_3 \\ \Delta\theta_4 \end{bmatrix}}_{\vec{\theta}} \tag{3.23}$$

Provided that $|J(\vec{\theta})| \neq 0$, we have

$$\begin{bmatrix} \Delta\theta_1 \\ \Delta\theta_2 \\ \Delta\theta_3 \\ \Delta\theta_4 \end{bmatrix} = J(\vec{\theta})^{-1} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \\ \Delta p \end{bmatrix} \tag{3.24}$$

Each component of Jacobian matrix is shown in Table A.3.

3.2 Jacobian of 1st Order Approximation

By looking over components of the Jacobian matrix, we find it contains many trigonometric functions. Decreasing the trigonometric functions in number will cause more efficiency in terms of mechanism and time for calculation. Then we introduce the appropriate Jacobian matrix so that we may eliminate the trigonometric functions, provided that θ is limited in the neighborhood of θ_o: $\theta_s (= \theta - \theta_o)$ is small.

At first we consider the 0th order approximation: $\sin\theta_s$ and $\cos\theta_s$ are expressed as $\sin\theta_s = 0$ and $\cos\theta_s = 1$, that is,

$$\left.\begin{aligned} \sin\theta &= \sin(\theta_o + \theta_s) \doteq \sin\theta_o \\ \cos\theta &= \cos(\theta_o + \theta_s) \doteq \cos\theta_o \end{aligned}\right\} \tag{3.25}$$

The 0th order approximation makes Jacobian matrix constant so that we do not need estimate it after the manipulator starts moving as well as we get $\vec{\theta}$ with ease. But it causes much error.

The 1st order approximation: sin θ_s and cos θ_s are expressed as $\sin\theta_s = \theta_s$ and $\cos\theta_s = 1$, that is,

$$\left.\begin{aligned} \sin\theta &= \sin(\theta_o + \theta_s) = \sin\theta_o + \theta_s \cos\theta_o \\ \cos\theta &= \cos(\theta_o + \theta_s) = \cos\theta_o - \theta_s \sin\theta_o \end{aligned}\right\} \qquad (3.26)$$

This approximation has less complicated calculation than the conventional one and less error than the 0th order approximation. The final form we get is expressed as

$$\Delta\theta_n = \frac{(\Delta x, \Delta y, \Delta z, \Delta p)[C_n]\begin{bmatrix}\theta_{1s}\\ \theta_{2s}\\ \theta_{3s}\\ \theta_{4s}\\ 1\end{bmatrix}}{(D_1, D_2, D_3, D_4)\begin{bmatrix}\theta_{1s}\\ \theta_{2s}\\ \theta_{3s}\\ \theta_{4s}\\ 1\end{bmatrix}} = \frac{\vec{\Delta x}^{-1}[C_n]\vec{\theta}_s}{\vec{D}^{-1}\vec{\theta}_s} \qquad (3.27)$$

where

$$\vec{\theta}_s = \vec{\theta} - \vec{\theta}_o$$

$$J(\vec{\theta}) \doteq \vec{D}^{-1} \cdot \theta_s$$

$[C_n]$ is constant

Next, as an example of higher order approximation, we consider the 2nd order approximation $\sin\theta_s$ and $\cos\theta_s$ are expressed as $\sin\theta_s = \theta_s$ and $\cos\theta_s = 1 - \frac{\theta_s^2}{2}$, that is,

$$\left.\begin{aligned}\sin\theta &= \sin(\theta_o + \theta_s) = (1 - \frac{\theta_s^2}{2})\sin\theta_o + \theta_s\cos\theta_o \\ \cos\theta &= \cos(\theta_o + \theta_s) = (1 - \frac{\theta_s^2}{2})\cos\theta_o - \theta_s\sin\theta_o\end{aligned}\right\} \tag{3.28}$$

These equations prove that the higher order approximation needs more complicated calculation compared with the 0th and 1st order ones, though it is more accurate. Accordingly, we conclude that the 1st order is the most appropriate in terms of simplicity and accuracy. The components of the 1st order approximation are expressed as

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & x_{13} & x_{14} & x_{15} \\ \vdots & & & & \vdots \\ x_{41} & \cdots & & \cdots & x_{45} \end{bmatrix} \begin{bmatrix} \theta_{1s} \\ \theta_{2s} \\ \theta_{3s} \\ \theta_{4s} \\ 1 \end{bmatrix} \tag{3.29}$$

$$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} y_{11} & \cdots & y_{15} \\ \vdots & & \vdots \\ y_{41} & \cdots & y_{45} \end{bmatrix} \begin{bmatrix} \theta_{1s} \\ \theta_{2s} \\ \theta_{3s} \\ \theta_{4s} \\ 1 \end{bmatrix} \tag{3.30}$$

$$\begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} z_{11} & \cdots & z_{15} \\ \vdots & & \vdots \\ z_{41} & \cdots & z_{45} \end{bmatrix} \cdot \begin{bmatrix} \theta_{1s} \\ \theta_{2s} \\ \theta_{3s} \\ \theta_{4s} \\ 1 \end{bmatrix} \tag{3.31}$$

$$\begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{bmatrix} = \begin{bmatrix} p_{11} & \cdots & p_{15} \\ \vdots & & \vdots \\ p_{41} & \cdots & p_{45} \end{bmatrix} \cdot \begin{bmatrix} \theta_{1s} \\ \theta_{2s} \\ \theta_{3s} \\ \theta_{4s} \\ 1 \end{bmatrix} \tag{3.32}$$

where $x_1 = \partial x/\partial\theta_1$.

Details are shown in Table A.4. Consequently, we have θ, through $\Delta\theta_4$

$$\Delta\theta_n = \Delta_n/\Delta \quad (n = 1, \cdots 4)$$

where

$$\Delta = \begin{vmatrix} x_1 & x_2 & \cdots & x_4 \\ y_1 & & & \vdots \\ z_1 & & & \vdots \\ p_1 & \cdots & & p_4 \end{vmatrix} \tag{3.33}$$

$$\Delta n = \begin{array}{c} \ \overbrace{}^{n} \\ \begin{vmatrix} x_1 & \Delta x & \cdots & x_4 \\ y_1 & \Delta y & & \vdots \\ z_1 & \Delta z & & \vdots \\ p_1 & \Delta p & \cdots & p_4 \end{vmatrix} \end{array} \tag{3.34}$$

In reference to θ_5 and θ_6, they are much influenced by the posture of the hand so that they change more than others. Therefore, introducing the approximate calculus for θ_5 and θ_6 is not suitable. So we get θ_5 and θ_6 by the basic expression.

$$\theta_5 = f(\theta_1, \theta_2, \theta_3, \theta_4), \ \theta_6 = f(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5) \qquad (3.35)$$

According to those methods, we can get $(\theta_1, \theta_2, - - - - -, \theta_6)$, consequently x, y, z and S by means of equations (3.15) - (3.18).

As the result, we can control $\vec{A}$.

3.3 Compensation

According to Chapter 2, $\vec{T}$ is measured by the measurement arm. We show the basic idea of measuring the position and orientation of the table. The measurement arm consisted of 6 joints all of which are rotational pairs. In practical use, the extreme distal link is kept perpendicular to the horizontal plane. On this condition, we can get x, y and z coordinates of point Q and angle α. For convenience, we define angles θ_1', θ_2' and length ℓ as in Fig. 3.4 and Fig. 3.5.

Accordingly, point Q is determined by

$$\left.\begin{aligned} Q_X &= \ell \cos\theta_2' \\ Q_Y &= \ell \sin\theta_2' \\ Q_Z &= \ell \sin\theta_1' - (b \cos(\theta_1' + \theta_4) + C) \end{aligned}\right\} \qquad (3.36)$$

where

$$\ell = a \cos \theta_1' + b \sin (\theta_1' + \theta_4)$$

$$\theta_1' = \tan^{-1} (1/(\tan\theta_1 \tan\theta_2))$$

$$\theta_2' = -\tan^{-1} (\tan\theta_2/\sin\theta_1)$$

These coordinates are determined in terms of the coordinate system which is fixed in the measurement arm. However, they need to be expressed by the common coordinate system which the slave arm can use. In terms of the common system, point Q is expressed by

$$\left.\begin{aligned} Q_x &= Q_X + \Delta x \\ Q_y &= Q_Y + \Delta y \\ Q_z &= Q_Z + \Delta z \end{aligned}\right\} \quad . \tag{3.37}$$

Where $(\Delta x, \Delta y, \Delta z)$ means the vector which shows the distance between two origins of coordinate system.

$$\alpha = \frac{\pi}{2} - (\theta_2' - \theta_6) \tag{3.38}$$

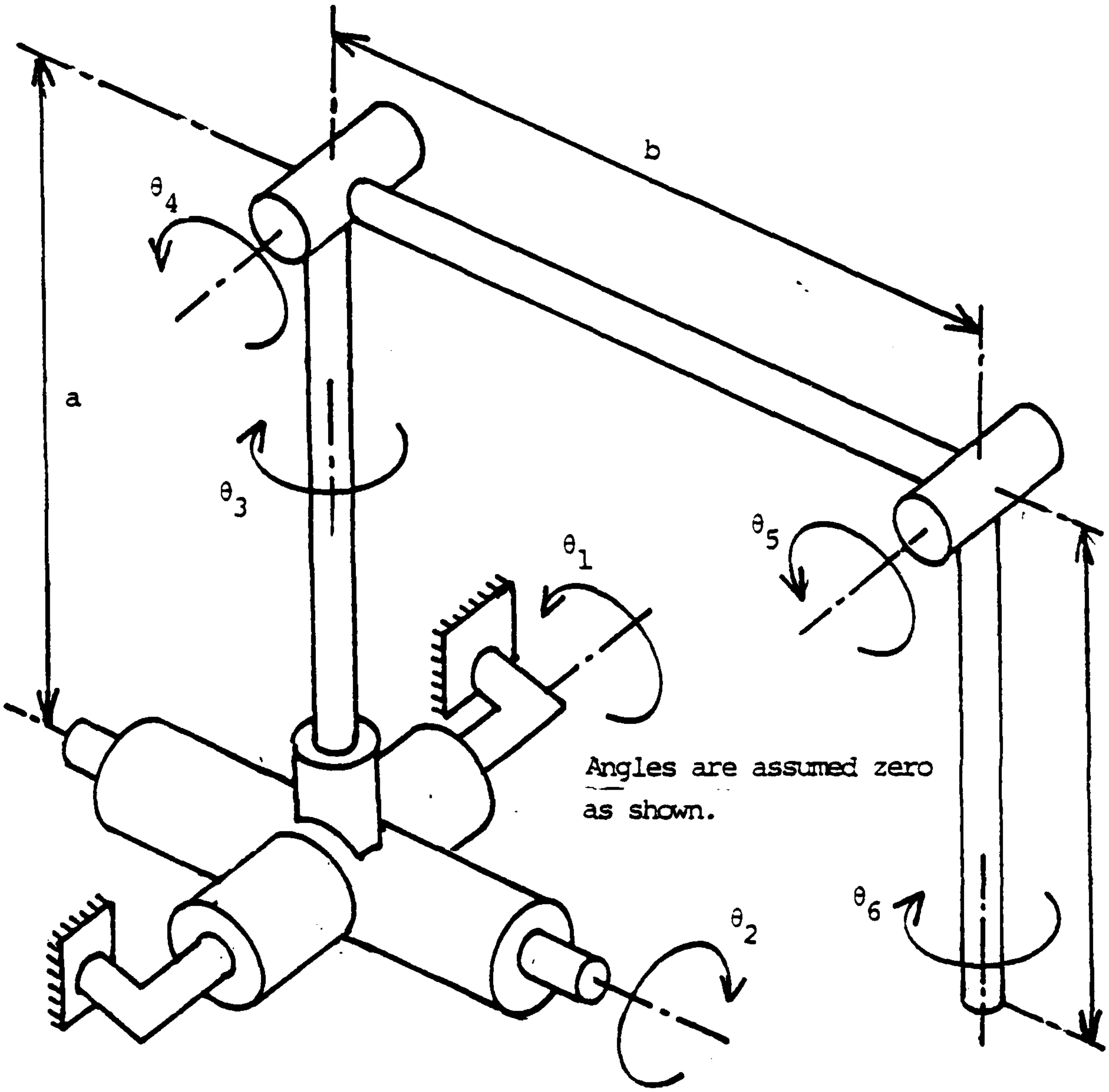
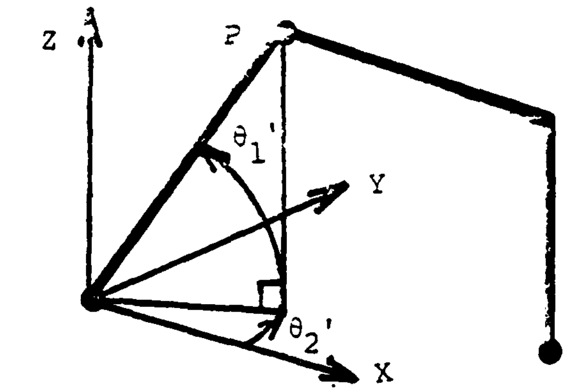
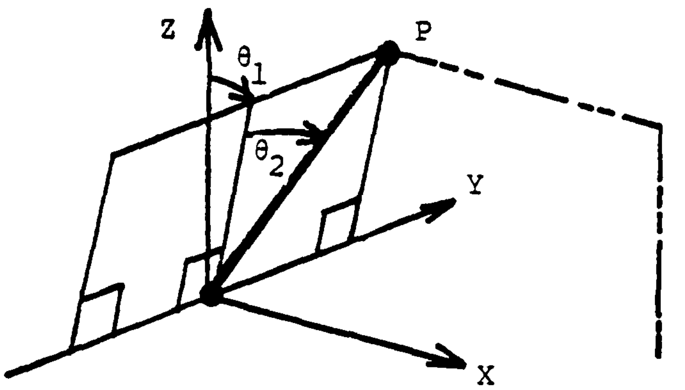


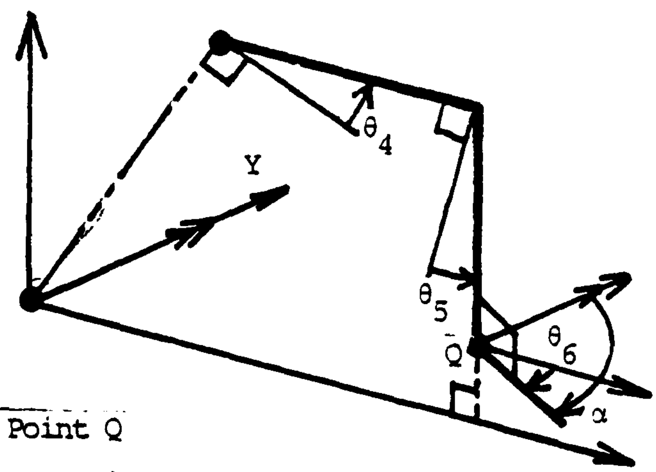
Fig. 3.3 Rotational Angles of the Measurement Arm



a) θ_1' and θ_2'

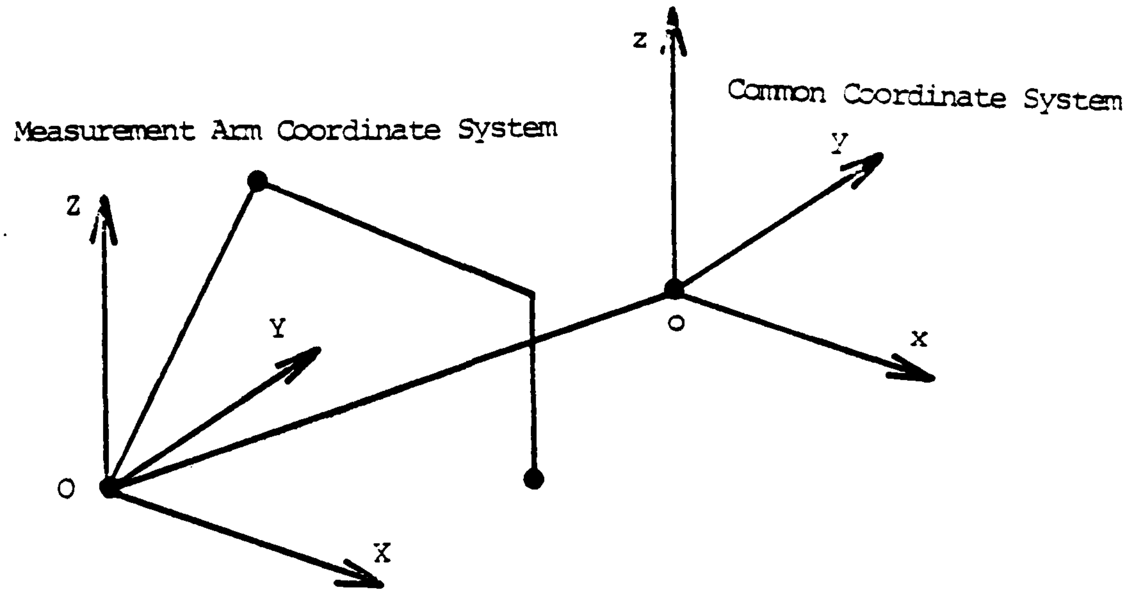


b) Point P

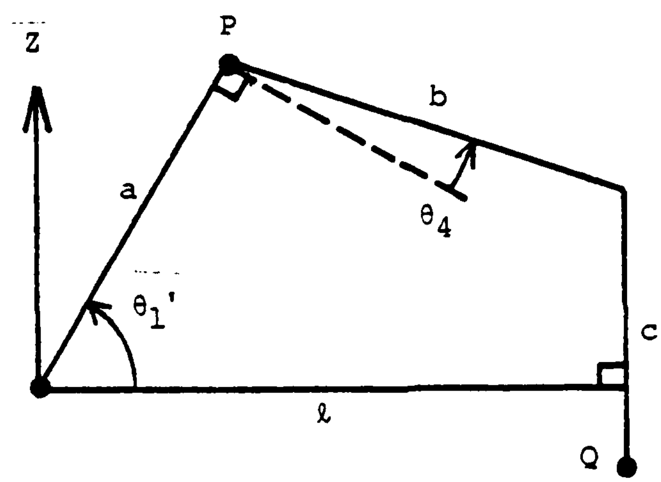


c) Point Q

Fig. 3.4 Rotational Angles in Practical Use



a) Two Coordinate Systems



b) Plane including all Links of Measurement Arm

Fig. 3.5 Rotational Angles and Some Other Parameters

CHAPTER 4

EXPERIMENTS

We mainly had two experiments: one was to measure the accuracy of the Measurement Arm; another was to measure the accuracy of the first order of approximation of Jacobian Matrices used for the straight-line motion of Slave Arm.

As to the first one, we combined the Measurement Arm with the Table illustrated in Fig. 2.1.b. and made the table move with the distance of $\Delta\vec{x}(\ell x, \ell y, \ell z)$ by means of the program MSURE. As illustrated in Figs. 4.1 and 4.2, we got good linearity, with errors within ± 0.15 inch. This is a good result, considering the experimental mechanism of the Measurement Arm.

Next, we tried to make a straight-line motion of the Slave Arm by means of program MAIN. As illustrated in Fig. 4.3, we make the arm move from origin to point p_1, the distance $\vec{x}_1$. The results are shown in Tables 4.3, 4.4 and 4.5. For example, as to the reference $\vec{x}_1(\Delta x_1, 0, 0)$ in Table 4.3, we have the errors within 0.09 inch, 0.19 inch and 0.21 inch, along the x-axis, y-axis and z-axis, respectively. This is also a good result, considering that the Slave Arm mechanism has significant backlash. The errors along the y-axis are big compared with others. It is mainly caused by the backlash of the angle θ_1.

In Figs. 4.1.c. and 4.2.c. we can see the increase in error, it is because the z-axis of the Measurement Arm is not in parallel with the z-axis of the Table.

TABLE 4.1

ACCURACY OF MEASUREMENT ARM

a) Measured Data $\ell_y = 0$, $\ell_z = 0$

ℓ_x inch	x inch	y inch	z inch	S
0	4.71	-6.78	3.89	0.870
1	5.75	-6.84	3.96	0.879
2	6.73	-6.85	4.03	0.890
3	7.61	-6.78	4.10	0.908
4	8.53	-6.75	4.14	0.932
5	9.49	-6.67	4.19	0.964

b) Modified Data $\ell_y = 0$, $\ell_z = 0$

ℓ_x inch	x inch	y inch	z inch	S
0	0.00	0.00	0.00	0.000
1	1.04	-0.06	0.07	0.009
2	2.02	-0.07	0.14	0.020
3	2.90	0.00	0.21	0.038
4	3.82	0.03	0.25	0.062
5	4.78	0.11	0.30	0.094

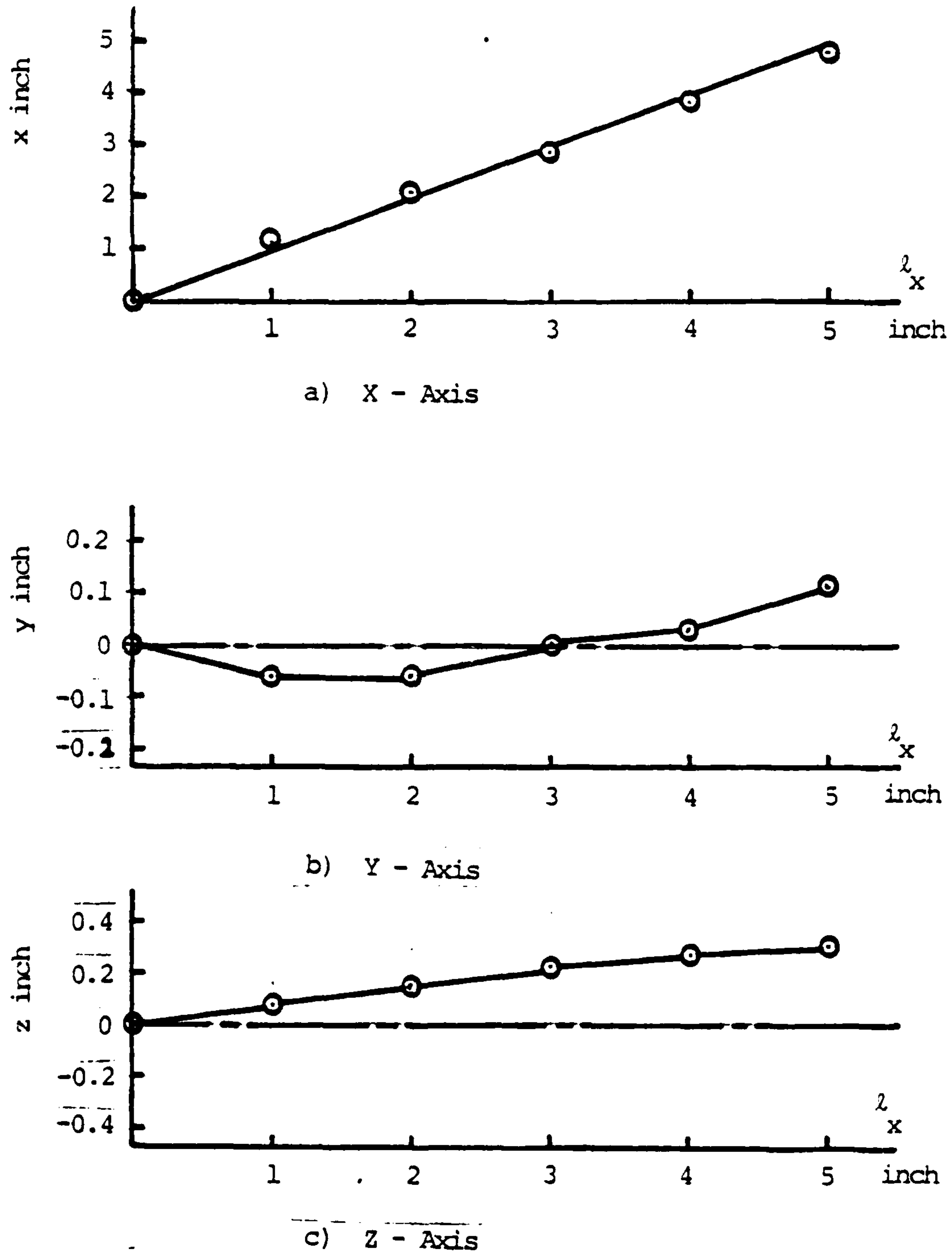


Fig. 4.1 Accuracy of Measurement Arm, $\ell_y = 0$, $\ell_z = 0$

TABLE 4.2

ACCURACY OF MEASUREMENT ARM

a) Measured Data $\ell_x = 0$, $\ell_z = 0$

ℓ_y inch	x inch	y inch	z inch	S
0	3.74	-12.71	3.70	0.812
1	3.71	-11.79	3.74	0.815
2	3.74	-10.80	3.77	0.807
3	3.79	- 9.76	3.79	0.811
4	3.83	- 8.71	3.80	0.823
5	3.84	- 7.77	3.80	0.842

b) Modified Data $\ell_x = 0$, $\ell_z = 0$

ℓ_y inch	x inch	y inch	z inch	S
0	0.00	0.00	0.00	0.000
1	-0.03	0.92	0.04	0.003
2	0.00	1.91	0.07	-0.005
3	0.05	2.95	0.09	-0.001
4	0.09	4.00	0.10	0.011
5	0.10	4.94	0.10	0.030

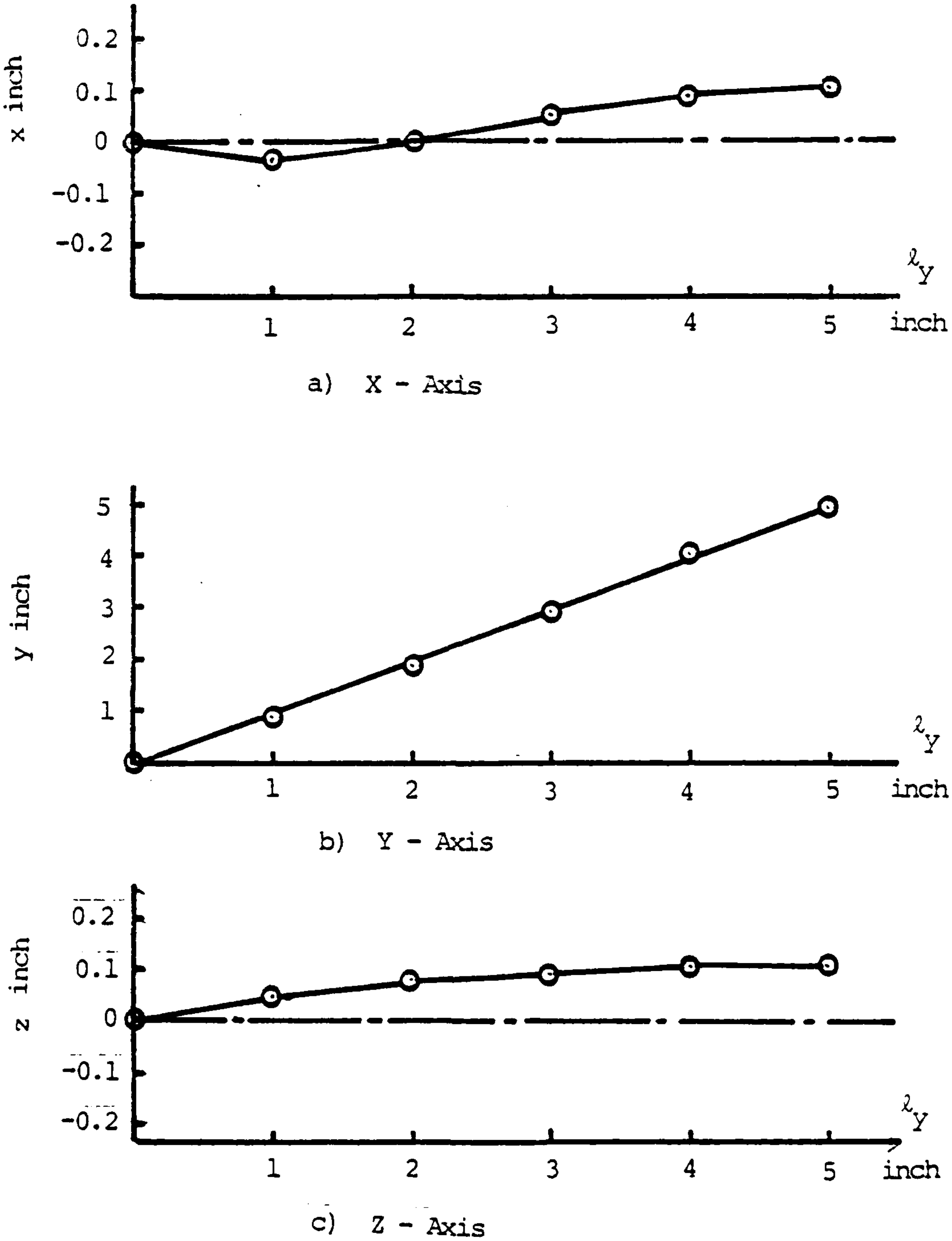


Fig. 4.2 Accuracy of Measurement Arm, $\ell_x = 0$, $\ell_z = 0$

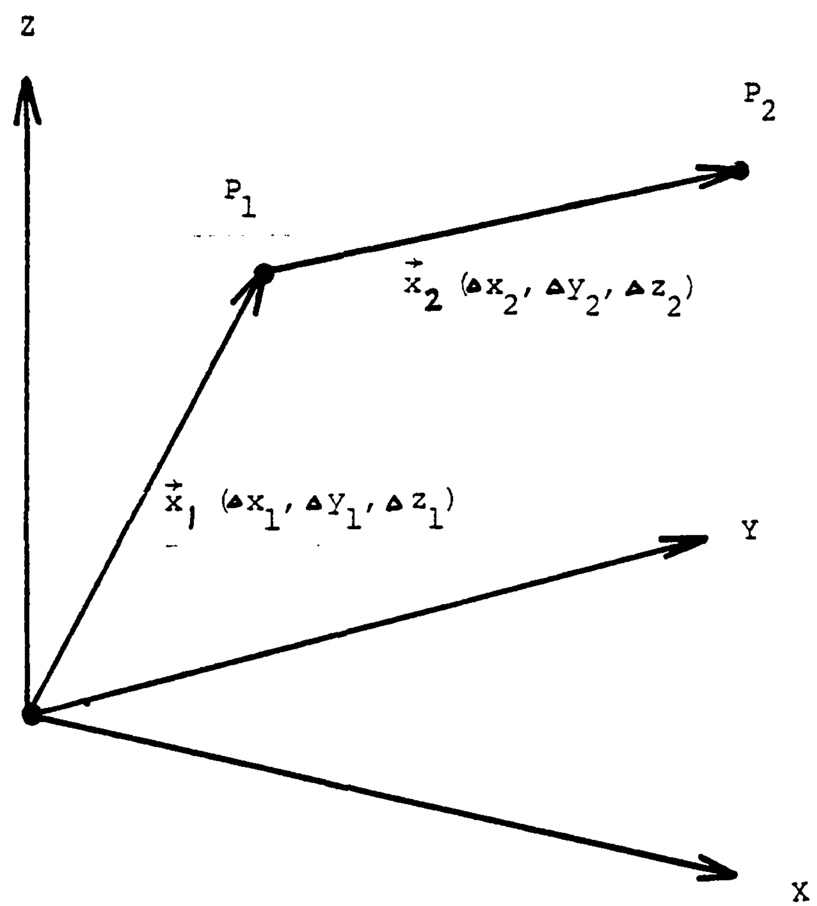


Fig. 4.3 Accuracy of Approximation Vehicle Frame

TABLE 4.3

Accuracy of Approximation $\vec{x}_1$, $\Delta y_1 = \Delta z_1 = 0$

a) Measured Data

inch

Δx_1	x	y	z
0	-0.01	-0.13	-0.00
0.5	0.49	-0.35	-0.04
1	0.98	-0.17	-0.03
2	1.98	-0.32	-0.05
3	2.99	-0.32	0.13
4	3.90	-0.32	0.21

b) Modified Data

inch

Δx_1	x	y	z
0	0.00	0.00	0.00
0.5	0.50	-0.22	-0.04
1	0.99	-0.04	-0.03
2	1.99	-0.19	-0.05
3	3.00	-0.19	0.13
4	3.91	-0.19	0.21

TABLE 4.4

Accuracy of Approximation $\vec{x}_1$, $\Delta x_1 = \Delta z_1 = 0$

a) Measured Data

inch

Δy_1	x	y	z
0	-0.00	-0.06	-0.01
0.5	-0.02	0.32	-0.03
1	-0.07	0.88	-0.09
2	-0.02	1.74	-0.15
3	-0.07	2.84	0.19
4	-0.06	3.87	0.26

b) Modified Data

inch

Δy_1	x	y	z
0	0.00	0.00	0.00
0.5	-0.02	0.38	-0.02
1	-0.07	0.94	-0.08
2	-0.02	1.80	-0.14
3	-0.07	2.90	0.20
4	-0.06	3.93	0.27

TABLE 4.5

Accuracy of Approximation $\vec{x}_1$, $\Delta x_1 = \Delta y_1 = 0$

a) Measured Data

inch

Δz_1	x	y	z
0	-0.00	-0.06	-0.01
0.2	0.03	-0.22	0.13
0.5	-0.04	-0.16	0.45
1	0.01	-0.31	0.97
2	-0.05	-0.30	1.97
3	0.01	-0.55	2.90

b) Modified Data

inch

Δz_1	x	y	z
0	0.00	0.00	0.00
0.2	0.03	-0.16	0.14
0.5	-0.04	-0.10	0.46
1	0.01	-0.25	0.98
2	-0.05	-0.24	1.98
3	0.01	-0.49	2.91

TABLE 4.6

Accuracy of Approximation $\vec{x}_2$, $y_2 = z_2 = 0$

a) Measured Data

inch

Δx_2	x	y	z
0	2.92	-2.22	1.15
0.5	3.46	-2.50	1.20
1	3.93	-2.19	1.14
2	4.90	-2.32	1.28
3	5.82	-2.26	1.36
4	6.88	-2.43	1.63

b) Modified Data

inch

Δx_2	x	y	z
0	0.00	0.00	0.00
0.5	0.54	-0.28	0.05
1	1.01	0.03	-0.01
2	1.98	-0.10	0.13
3	2.90	-0.04	0.21
4	3.96	-0.21	0.48

TABLE 4.7

Accuracy of Approximation $\vec{x}_2$, $\Delta x_2 = \Delta z_2 = 0$

a) Measured Data

inch

Δy_2	x	y	z
0	2.92	-2.12	1.14
0.5	2.94	-1.84	1.09
1	2.93	-1.43	1.10
2	3.00	-0.48	1.11
3	3.00	0.73	1.05
4	2.88	1.77	1.16

b) Modified Data

inch

Δy_2	x	y	z
0	0.00	0.00	0.00
0.5	0.02	0.28	-0.05
1	0.01	0.69	-0.04
2	0.08	1.64	-0.03
3	0.08	2.85	-0.09
4	-0.04	3.89	0.02

TABLE 4.8

Accuracy of Approximation $\vec{x}_2$, $\Delta x_2 = \Delta y_2 = 0$

a) Measured Data

inch

Δz_2	x	y	z
0	3.00	-2.32	1.19
0.5	2.87	-2.25	1.58
1	2.96	-2.30	2.05
2	2.92	-2.64	3.10
3	2.86	-2.81	4.08
4	2.82	-3.06	5.04

b) Modified Data

inch

Δz_2	x	y	z
0	0.00	0.00	0.00
0.5	-0.14	0.07	0.39
1	-0.04	0.02	0.86
2	-0.08	-0.32	1.91
3	-0.14	-0.49	2.89
4	-0.18	-0.74	3.85

CHAPTER 5

CONCLUSIONS AND RECOMMENDATIONS FOR FUTURE WORK

The purpose of this research was to extend the compensation done previously by Tani which presupposed a perfect measurement by using a sensor.

For this purpose, as a sensor we developed a 6 degree-of-freedom passive Measurement Arm having a simple gripper but otherwise flaccid. In addition to developing the sensor, we extended the control method by considering some approximated Jacobian matrices instead of complicated strict Jacobian matrices. Consequently, we developed the Jacobian matrices of first order approximation with no trigonometric functions.

We performed two experiments: one was to measure the accuracy of the Measurement Arm by moving the table; another was to measure the accuracy of the Jacobian matrices of first order approximation by moving the Slave Arm. The result showed the capability of compensation with the Measurement Arm as a sensor.

Having developed the basic hardware and software for compensation by means of the sensor, we intend to improve this compensation by using these techniques in the near future.

As an example of future work, I show the following two programs. One is the program named TTT, which will keep the same orientation of the hand of the Slave Arm and the same distance between the table and Slave Arm, regardless of the position of the table. This is to test the total error caused by the hardware and software of the system. Another is the program named HICOM which is to extend Tani's compensation by using the

sensor. These are almost completed but not yet perfect.

In order to extend the compensation with sensor, we suggest developing a program which considers the errors caused by mechanisms such as backlash in addition to present considerations.

REFERENCES

1. Sheridan, T.B. and Verplank, W.L., "Human and Computer Control of Undersea Teleoperators", Man-Machine Systems Laboratory Report, Massachusetts Institute of Technology, July 1978.

2. Tani, K., "Supervisory Control of Remote Manipulation with Compensation for Moving Target", Man-Machine Systems Laboratory Report, Massachusetts Institute of Technology, July 1980.

3. Crandall, S.H., et.al., "Dynamics of Mechanical and Electromechanical Systems", McGraw-Hill Book Company, 1968.

4. Brooks, T.L., "Superman: A System for Supervisory Manipulation and the Study of Human/Computer Interactions", Master's Thesis, Massachusetts Institute of Technology, May 1979.

5. Hildebrand, F.B., "Advanced Calculus for Applications", Prentice-Hall, Inc. Englewood Cliffs, New Jersey, 1976.

6. Witney, D.E., "Resolved Motion Rate Control of Manipulators and Human Prostheses", IEEE Transactions on Man-Machine Systems, Vol. MMS-10, No. 2, June 1969.

APPENDIX I

TRANSFORMATION MATRICES

TABLE A.1

Frame Transformation

$$
{}^{0}A_{1} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta_1 & -\sin\theta_1 & 0 \\ 0 & \sin\theta_1 & \cos\theta_1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{1}A_{2} = \begin{bmatrix} \cos\theta_2 & 0 & \sin\theta_2 & 0 \\ 0 & 1 & 0 & 18 \\ -\sin\theta_2 & 0 & \cos\theta_2 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{2}A_{3} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta_3 & -\sin\theta_3 & 0 \\ 0 & \sin\theta_3 & \cos\theta_3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{3}A_{4} = \begin{bmatrix} \cos\theta_4 & -\sin\theta_4 & 0 & -1.39\sin\theta_4 \\ \sin\theta_4 & \cos\theta_4 & 0 & 1.39\cos\theta_4 \\ 0 & 0 & 1 & -40 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{4}A_{5} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\theta_5 & -\sin\theta_5 & 0 \\ 0 & \sin\theta_5 & \cos\theta_5 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

$$
{}^{5}A_{6} = \begin{bmatrix} \cos\theta_6 & 0 & \sin\theta_6 & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta_6 & 0 & \cos\theta_6 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

TABLE A.2

Transformation from Hand to Vehicle

$$
{}^6A_0 = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

a_{11} = (C2C4 + S2S3S4) C6 + (S2S3C4S5 - C2S4S5 - S2C3C5) 56

a_{12} = (S2S3C4 - C2S4) C5 + S2C3S5

a_{13} = (C2S4S5 - S2S3C4S5 + S2C) C6 + (C2C4 + S2S3S4) S6

a_{14} = 1.39 (S2S3C4 - C2S4) - 40 (S2C3)

a_{21} = (C1C3C4 - S1C2S3C4 - S1S2S4) S5S6 + (C1S3 + S1C2C3) C5S6 + (C1C3 - S1C2S3)S4C6 + S1S2C4C6

a_{22} = (C1C3C4 - S1S2S4 - S1C2S3C4) C5 - (C1S3 + S1C2C3) S5

a_{23} = (S1S2S4 - C1C3C4 + S1C2S3C4) S5C6 - (C1S3 + S1C2C3) C5C6 + (C1C3 - S1C2S3) S4S6 + S1S2C4S6

a_{24} = 1.39 (C1C3C4 - S1C2S3C4 - S1S2S4) + 40 (C1S3

a_{31} = (C1S2S4 + S1C3C4 + C1C2S3C4)S5S6 + (S1S3 - C1S2C3) C5 + (C1C2S3 + S1C3) S4C6 - C1S2C4C6

a_{32} = (C1S2S4 + S1C3C4 + C1C2S3C4) C5 + (C1C2C3 - S1S3)

a_{33} = -(C1S2S4 + S1C3C4 + C1C2S3C4) S5C6 + (S1C3 + C1C2S3) +(C1C2C3 - S1S3) C5C6 - C1S2C4S6

a_{34} = 1.39 (S1C3C4 + C1C2S3C4 + C1S2S4) + 40 (S1S3 - C1C2C3) + 18S1

where S1 = $\sin\theta_1$, C1 = $\cos\theta_1$

APPENDIX II

JACOBIAN MATRICES

TABLE A.3

Jacobian Matrix

$$J(\vec{\theta}) = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \\ z_1 & z_2 & z_3 & z_4 \\ p_1 & p_2 & p_3 & p_4 \end{bmatrix}$$

where $x_1 = \partial x/\partial\theta_1$.

$x_1 = 0$

$x_2 = 1.39\ (C2S3C4 + S2S4) - 40C2C3$

$x_3 = 1.39S2C3C4 + 40S2S3$

$x_4 = -1.39\ (C2C4 + S2S3S4)$

$y_1 = -1.39\ (S1C3C4 + C1C2S3C4 + C1S2S4)$
$\quad + 40\ (C1C2C3 - S1S3) - 18S1$

$y_2 = 1.39\ (S1S2S3C4 - S1C2S4) - 40S1S2C3$

$y_3 = -1.39\ (S1C2C3C4 + C1S3C4) + 40\ (C1C3 - S1C2S3)$

$y_4 = 1.39\ (-S1S2C4 - C1C3S4 + S1C2S3S4)$

$z_1 = 1.39\ (C1C3C4 - S1C2S3C4 - S1S2S4)$
$\quad + 40\ (S1C2C3 + C1S3) + 18C1$

$z_2 = 1.39\ (-C1S2S3C4 + C1C2S4) + 40C1S2C3$

$z_3 = 1.39\ (C1C2C3C4 - S1S3C4) + 40\ (S1C3 + C1C2S3)$

$z_4 = 1.39\ (C1S2C4 - S1C3S4 - C1C2S3S4)$

$p_1 = 0 + C1S2C4 - S1C3S4 - C1C2S3S4$

$p_2 = S\ (-S2C4 + C2S3S4) + (S1C2C4 + S1S2S3S4)$

$p_3 = S\ (S2C3S4) + (-S1C2C3S4 - C1S3S4)$

Table A.3 (Continued)

$$P_4 = S\,(S2S3C4 - C2S4) + (C1C3C4 - S1C2S3C4 - S1S2S4)$$

where $S1 = \sin\theta_1$ and $C1 = \cos\theta_1$.

TABLE A.4

Components of Matrix

$x_{11} = 0$

$x_{12} = 0$

$x_{13} = 0$

$x_{14} = 0$

$x_{15} = 0$

$x_{21} = 0$

$x_{22} = 1.39\ [(-S30 - C40)\ S20 + S40C20] + 40C30S20$

$x_{23} = 1.39\ (C20 + C40)\ C30 + 40C20S30$

$x_{24} = 1.39\ [(-C20 - S30)\ S40 + S20C40]$

$x_{25} = 1.39\ (C20S30C40 + S20S40) - 40C20C30$

$x_{31} = 0$

$x_{32} = 1.39\ (C20C30C40) + 40C20S30$

$x_{33} = -1.39\ S20S30C40 + 40S20C30$

$x_{34} = -1.39S20C30S40$

$x_{35} = 1.39S20C30C40 + 40S20S30$

$x_{41} = 0$

$x_{42} = 1.39\ (S20C40 - C20S30S40)$

$x_{43} = 1.39S20C30S40$

$x_{44} = 1.39\ (C20S40 - S20S30C40)$

$x_{45} = -1.39\ (C20C40 + S20S30S40)$

where $C10 = \cos\theta_{10}$, $S10 = \sin\theta_{10}$

Table A.4 (Continued)

y_{11} = -1.39 (C10C30C40 - S10C20S30C40)
+ 40 (-S10C20C30 - C10S30) - 18C10

y_{12} = -1.39 [C10 (-S20) S30 C40] + 40 [C10 (-S20) C30]

y_{13} = -1.39 [S10 (-S30) C40 + C10C20C30C40]
+ 40 [C10C20 (-S30) - S10C30)

y_{14} = -1.39 [S10C30 (-S40) + C10C20S30 (-S40)]

y_{15} = -1.39 (S10C30C40 + C10C20S30C40)
+ 40 (C10C20C30 - S10S30) - 18S10

y_{21} = 1.39 (C10S20S30C40 - C10C20S40)
- 40C10S20C30

y_{22} = 1.39 [S10C20S30C40 - S10 (-S20) S40]
- 40S10C20C30

y_{23} = 1.39S10S20C30C40 - 40S10S20 (-S30)

y_{24} = 1.39 [S10S20S30 (-S40) - S10C20C40]

y_{25} = 1.39 (S10S20S30C40 - S10C20S40)
- 40S10S20C30

y_{31} = -1.39 (C10C20C30C40 - S10S30C40)
+ 40 (-S10C30 - C10C20S30)

y_{32} = -1.39 [S10 (-S20) C30C40] - 40S10 (-S20) S30

y_{33} = -1.39 [S10C20 (-S30) C40 + C10C30C40]
+ 40 [C10 (-S30) - S10C20C30)

y_{34} = -1.39 [S10C20C30 (-S40) + C10S30 (-S40)]

y_{35} = -1.39 (S10C20C30C40 + C10S30C40)
+40 (C10C30 - S10C20S30)

y_{41} = 1.39 (-C10S20C40 + S10C30S40
+ C10C20S30S40)

y_{42} = 1.39 [-S10C20C40 + S10 (-S20) S30S40]

y_{43} = 1.39 [-C10 (-S30) S40 + S10C20C30S40]

y_{44} = 1.39 [-S10S20 (-S40) - C10C30C40
+ S10C20S30C40]

y_{45} = 1.39 (-S10S20C40 - C10C30S40
+ S10C20S30S400

z_{11} = 1.39 (-S10C30C40 - C10C20S30C40 - C10S20S40)
+ 40 (C10C20C30 - S10S30) + 18 (-S10)

z_{12} = 1.39 [-S10 (-S20) S30C40 - S10C20S40]
+ 40 [S10 (-S20) C30]

z_{13} = 1.39 [C10 (-S30) C40 - S10C20C30C40]
+ 40 [S10C20 (-S30) + C10C30]

z_{14} = 1.39 [C10C30 (-S40) - S10C20S30 (-S40)
- S10S20C40

z_{15} = 1.39 (C10C30C40 - S10C20S30C40 - S10S20S40
+ 40 (S10C20C30 + C10S30) + 18C10

z_{21} = 1.39 (S10S20S30C40 - S10C20S40)
+ 40 (- S10S20C30)

z_{22} = 1.39 [-C10C20S30C40 + C10 (-S20) S40]
+ 40C10C20C30

z_{23} = 1.39 (-C10) S20C30C40 + 40C10S20 · (-S30)

z_{24} = 1.39 [-C10S20S30 (-S40) + C10C20C40]

z_{25} = 1.39 (-C10S20S30C40 + C10C20S40)
+ 40C10S20C30

z_{31} = 1.39 (-S10C20C30C40 - C10S30C40)
+ 40 (C10C30 - S10C20S30)

z_{32} = 1.39 C10 (-S20) C30C40 + 40C10 (-S20) S30

z_{33} = 1.39 [C10C20 (-S30) C40 - S10C30C40]
+ 40 [S10 (-S30) + C10C20C30]

z_{34} = 1.39 [C10C20C30 (-S40) - S10S30 (-S40)]

z_{35} = 1.39 (C10C20C30C40 - S10S30C400
+ 40 (S10C30 + C10C20S30)

Table A.4 (Continued)

z_{41} = 1.39 (-S10S20C40 - C10C30S40 + S10C20S30S40)

z_{42} = 1.39 [C10C20C40 - C10 (-S20) S30S40]

z_{43} = 1.39 [-S10 (-S30) S40 - C10C20C30S40]

z_{44} = 1.39 [C10S20 (-S40) - S10C30C40 - C10C20S30C40]

z_{45} = 1.39 (C10S20C40 - S10C30S40 - C10C20S30S40)

p_{11} = -S10S20C40 - C10C30S40 + S10C20S30S40

p_{12} = C10C20C40 - C10 (-S20) S30S40

p_{13} = -S10 (-S30)S40 - C10C20C30S40

p_{14} = C10S20 (-S40) - S10C30C40 - C10C20S30C40

p_{15} = C10S20C40 - S10C30S40 - C10C20S30S40

p_{21} = C10C20C40 + C10S20S30S40

p_{22} = S [-C20C40 - S20S30S40]
+ S10 (-S20) C40 + S10C20S30S40

p_{23} = S [C20C30S40]
+ S10S20C30S40

p_{24} = S [-S20 (-S40) + C20S30C40]
+ S10C20 (-S40) + S10S20S30C40

p_{25} = S [-S20C40 + C20S30S40]
+ S10C20C40 + S10S20S30S40

p_{31} = -C10C20C30S40 + S10S30S40

p_{32} = S [C20C30S40]
- S10 (-S20) C30S40

p_{33} = S [S20 (-S30) S40]
- S10C20 (-S30) S40 - C10C30S40

p_{34} = S [S20C30S40]

p_{35} = S [S20C30S40]
- S10C20C30S40 - C10S30S40

Table 4.3 (Continued)

$$P_{41} = - S10C30C40 - C10C20S30C40 - C10S20S40$$

$$P_{42} = S\ [C20S30C40 + S20S40]$$

$$P_{43} = S\ [S20C30C40] + C10\ (-S30)\ C40 - S10C20C30C40$$

$$P_{44} = S\ [S20S30\ (-S40) - C20C40] + C10C30\ (-S40) - S10C20S30\ (-S40) - S10S20C40$$

$$P_{45} = S\ [S20S30C40 - C20S40] + C10C30C40 - S10C20S30C40 - S10S20S40$$

APPENDIX III

LIST OF COMPUTER PROGRAMS

"MAIN": Straight-line motion of Slave Arm according to the reference (Δx, Δy, Δx, $\Delta\alpha$) in Figs. 4.3 and 3.2, by means of Jacobian Matrices of 1st Order Approximation in Secs. 3.1 and 3.2.

"MSURE": Measurement of Point Q (x, y, z, α) in Fig. 3.4 which is the extreme end of Measurement Arm, by means of the method in Sec. 3.3.

"TTT": Motion of Slave Arm so as to keep the constant distance between the table measured by "SUBME" and Slave Arm measured by "SUBS1", regardless of the position of table.

"HICOM": Extended Compensation done so far by means of Measurement Arm as a sensor.

Subroutines for "TTT" and "HICOM"

"SUBI", "SUBO": Input and Output of each angle.

"SUBS1" and "SUBM1": Present angle of Slave Arm and Master Arm.

"SUBS2" and "SUBM2": Desired angle according to the reference (Δx, Δy, Δz, $\Delta\alpha$) for Slave Arm and for Master Arm.

"SUBME": Measurement of the extreme end of Measurement Arm.

```
      PROGRAM MAIN
C     MAIN
      COMMON THXSI(7),THXMI(7),THXSO(7),THXMO(7)
      COMMON IDATA(14)
      DIMENSION AD(7),PX(4,5),PY(4,5),XY(4,5),YY(4,5),ZZ(4,5),PP(4,5),
     1        X(4),Y(4),Z(4),P(4),TSS(5)
C
C
      THXSO(1)=0.0
      THXMO(1)=0.0
      CALL ANINIT
      CALL DOUT(24,0)
      CALL DOUT(26,0)
      TYPE *,'MANIPULATOR COMPUTER CONTROL'
      ACCEPT *,I
      IF(I.NE.1) STOP
      CALL AINSQ(16,29,IDATA)
      CALL AOUTSQ(4,17,IDATA)
      CALL DOUT(24,63)
C
10    CONTINUE
C
20    CONTINUE
      TYPE *,'DO YOU NEED ORIGIN SET? /1'
      ACCEPT *,NNN
      IF(NNN.EQ.1) GO TO 500
C
      TYPE *,'NO.OF.DIV : SPEED'
      ACCEPT *,N
      RN=FLOAT(N)
C
      TYPE *,'POSITION INCREMENTS DX,DY,DZ,IN INCH AND IS,IN DEG'
      ACCEPT *,DX,DY,DZ,IS
      SS=FLOAT(IS)*(3.1416/180.0)
      SS=TAN(SS)
C
C
      CALL MANGLI
C
      THS1=THXSI(5)
      THS2=THXSI(7)
      THS3=THXSI(6)-THS1
      THS4=THXSI(2)
      THS5=(THXSI(4)+THXSI(3))/2.0+0.27*THS3
      THS6=(THXSI(4)-THXSI(3))/1.65
C
      THM1=THXMI(5)
      THM2=THXMI(7)
      THM3=THXMI(6)-THM1
      THM4=THXMI(2)
      THM5=(THXMI(4)+THXMI(3))/2.0+0.27*THM3
      THM6=(THXMI(4)-THXMI(3))/1.65
C
      TYPE *,'IDATA(1)-(3)'
      TYPE *,IDATA(1),IDATA(2),IDATA(3)
      TYPE *,'THS1-3'
      TYPE *,THS1,THS2,THS3
C
C
      TYPE *,'PRESENT THS'
      ITHS1=ININT((180.0/3.1416)*THS1)
      ITHS2=ININT((180.0/3.1416)*THS2)
      ITHS3=ININT((180.0/3.1416)*THS3)
      ITHS4=ININT((180.0/3.1416)*THS4)
      ITHS5=ININT((180.0/3.1416)*THS5)
      ITHS6=ININT((180.0/3.1416)*THS6)
      TYPE *,ITHS1,ITHS2,ITHS3,ITHS4,ITHS5,ITHS6
```

```
C
      TYPE *,'X,Y,Z'
      XP=     1.39*(SIN(THS2)*SIN(THS3)*COS(THS4)
     1        -COS(THS2)*SIN(THS4))
     2     -40.0*(SIN(THS2)*COS(THS3))
C
      YP=     1.39*(COS(THS1)*COS(THS3)*COS(THS4)
     1        -SIN(THS1)*COS(THS2)*SIN(THS3)*COS(THS4)
     2        -SIN(THS1)*SIN(THS2)*SIN(THS4))
     3     +40.0*(SIN(THS1)*COS(THS2)*COS(THS3)
     4        +COS(THS1)*SIN(THS3))
     5     +18.0*(COS(THS1))
C
      ZP=     1.39*(SIN(THS1)*COS(THS3)*COS(THS4)
     1        +COS(THS1)*COS(THS2)*SIN(THS3)*COS(THS4)
     2        +COS(THS1)*SIN(THS2)*SIN(THS4))
     3     +40.0*(-COS(THS1)*COS(THS2)*COS(THS3)
     4         +SIN(THS1)*SIN(THS3))
     5       +18.0*SIN(THS1)
C
      YP=YP-19.39
      ZP=ZP+40.0
C
      TYPE *,XP,YP,ZP
      TYPE *,'IF INITIAL SET NEEDS,INPUT 1'
      ACCEPT *,I1
      IF(I1.NE.1) GO TO 100
C
      THS01=THS1
      THS02=THS2
      THS03=THS3
      THS04=THS4
C
      S1=SIN(THS01)
      S2=SIN(THS02)
      S3=SIN(THS03)
      S4=SIN(THS04)
C
      C1=COS(THS01)
      C2=COS(THS02)
      C3=COS(THS03)
      C4=COS(THS04)
C
      XX(1,5)=0.0
      XX(1,1)=0.0
      XX(1,2)=0.0
      XX(1,3)=0.0
      XX(1,4)=0.0
C
      XX(2,5)=1.39*(C2*S3*C4+S2*S4)-40.0*(C2*C3)
      XX(2,1)=0.0
      XX(2,2)=1.39*((-S3-C4)*S2+S4*C2)+40.0*C3*S2
      XX(2,3)=1.39*(C2+C4)*C3-40.0*C2*S3
      XX(2,4)=1.39*((-C2-S3)*S4+S2*C4)
C
      XX(3,5)=1.39*(S2*C3*C4)+40.0*(S2*S3)
      XX(3,1)=0.0
      XX(3,2)=1.39*(C2*C3*C4)+40.0*(C2*S3)
      XX(3,3)=-1.39*(S2*S3*C4)+40.0*(S2*C3)
      XX(3,4)=-1.39*(S2*C3*S4)
C
      XX(4,5)=-1.39*(C2*C4+S2*S3*S4)
```

```
      XX(4,1)=0.0
      XX(4,2)=1.39*(S2*C4-C2*S3*S4)
      XX(4,3)=-1.39*(S2*C3*S4)
      XX(4,4)=1.39*(C2*S4-S2*S3*C4)
C
      YY(1,5)=-1.39*(S1*C3*C4+C1*C2*S3*C4)+40.0*(C1*C2*C3-S1*S3)
     1        -18.0*S1
      YY(1,1)=-1.39*(C1*C3*C4-S1*C2*S3*C4)+40.0*(-S1*C2*C3-C1*S3)
     1        -18.0*C1
      YY(1,2)=-1.39*(C2*(-S2)*S3*C4)+40.0*(C1*(-S2)*C3)
      YY(1,3)=-1.39*(S1*(-S3)*C4+C1*C2*C3*C4)
     1        +40.0*(C1*C2*(-S3)-S1*C3)
      YY(1,4)=-1.39*(S1*C3*(-S4)+C1*C2*S3*(-S4))
C
      YY(2,5)=1.39*(S1*S2*S3*C4-S1*C2*S4)-40.0*(S1*S2*C3)
      YY(2,1)=1.39*(C1*S2*S3*C4-C1*C2*S4)-40.0*(C1*S2*C3)
      YY(2,2)=1.39*(S1*C2*S3*C4-S1*(-S2)*S4)-40.0*(S1*C2*C3)
      YY(2,3)=1.39*(S1*S2*C3*C4)-40.0*(S1*S2*(-S3))
      YY(2,4)=1.39*(S1*S2*S3*(-S4)-S1*C2*C4)
C
      YY(3,5)=-1.39*(S1*C2*C3*C4+C1*S3*C4)+40.0*(C1*C3-S1*C2*S3)
      YY(3,1)=-1.39*(C1*C2*C3*C4-S1*S3*C4)+40.0*((-S1)*C3-C1*C2*S3)
      YY(3,2)=-1.39*(S1*(-S2)*C3*C4)-40.0*S1*(-S2)*S3
      YY(3,3)=-1.39*(S1*C2*(-S3)*C4+C1*C3*C4)+40.0*(C1*(-S3)
     1        -S1*C2*C3)
      YY(3,4)=-1.39*(S1*C2*C3*(-S4)+C1*S3*(-S4))
C
      YY(4,5)=1.39*(-S1*S2*C4-C1*C3*S4+S1*C2*S3*S4)
      YY(4,1)=1.39*(-C1*S2*C4+S1*C3*S4+C1*C2*S3*S4)
      YY(4,2)=1.39*(-S1*C2*C4+S1*(-S2)*S3*S4)
      YY(4,3)=1.39*(-C1*(-S3)*S4+S1*C2*C3*S4)
      YY(4,4)=1.39*(-S1*S2*(-S4)-C1*C3*C4+S1*C2*S3*C4)
C
C
      ZZ(1,5)=1.39*(C1*C3*C4-S1*C2*S3*C4-S1*S2*S4)
     1        +40.0*(S1*C2*C3+C1*S3)+18.0*C1
      ZZ(1,1)=1.39*(-S1*C3*C4-C1*C2*S3*C4-C1*S2*S4)
     1        +40.0*(C1*C2*C3-S1*S3)+18.0*(-S1)
      ZZ(1,2)=1.39*(-S1*(-S2)*S3*C4-S1*C2*S4)
     1        +40.0*(S1*(-S2)*C3)
      ZZ(1,3)=1.39*(C1*(-S3)*C4-S1*C2*C3*C4)
     1        +40.0*(S1*C2*(-S3)+C1*C3)
      ZZ(1,4)=1.39*(C1*C3*(-S4)-S1*C2*S3*(-S4)-S1*S2*C4)
C
C
      ZZ(2,5)=1.39*(-C1*S2*S3*C4+C1*C2*S4)
     1        +40.0*(C1*S2*C3)
      ZZ(2,1)=1.39*(S1*S2*S3*C4-S1*C2*S4)
     1        +40.0*(-S1*S2*C3)
      ZZ(2,2)=1.39*(-C1*C2*S3*C4+C1*(-S2)*S4)
     1        +40.0*(C1*C2*C3)
      ZZ(2,3)=1.39*(-C1*S2*C3*C4)+40.0*(C1*S2*(-S3))
      ZZ(2,4)=1.39*(-C1*S2*S3*(-S4)+C1*C2*C4)
C
C
      ZZ(3,5)=1.39*(C1*C2*C3*C4-S1*S3*C4)
     1        +40.0*(S1*C3+C1*C2*S3)
      ZZ(3,1)=1.39*(-S1*C2*C3*C4-C1*S3*C4)
     1        +40.0*(C1*C3-S1*C2*S3)
      ZZ(3,2)=1.39*(C2*(-S2)*C3*C4)+40.0*(C1*(-S2)*S3)
      ZZ(3,3)=1.39*(C1*C2*(-S3)*C4-S1*C3*C4)
     1        +40.0*(S1*(-S3)+C1*C2*C3)
      ZZ(3,4)=1.39*(C1*C2*C3*(-S4)-S1*S3*(-S4))
C
C
      ZZ(4,5)=1.39*(C1*S2*C4-S1*C3*S4-C1*C2*S3*S4)
      [illegible]
      [illegible]
      [illegible]
      ZZ(4,4)=1.39*(C1*S2*(-S4)-S1*C3*C4-C1*C2*S3*C4)
```

Copy available to DTIC does not permit fully legible reproduction

```
C
C
      PX(1,5)=0.0
      PX(1,1)=0.0
      PX(1,2)=0.0
      PX(1,3)=0.0
      PX(1,4)=0.0
C
      PX(2,5)=-S2*C4+C2*S3*S4
      PX(2,1)=0.0
      PX(2,2)=-C2*C4-S2*S3*S4
      PX(2,3)=C2*C3*S4
      PX(2,4)=-S2*(-S4)+C2*S3*C4
C
      PX(3,5)=S2*C3*S4
      PX(3,1)=0.0
      PX(3,2)=C2*C3*S4
      PX(3,3)=S2*(-S3)*S4
      PX(3,4)=S2*C3*C4
C
      PX(4,5)=S2*S3*C4-C2*S4
      PX(4,1)=0.0
      PX(4,2)=C2*S3*C4+S2*S4
      PX(4,3)=S2*C3*C4
      PX(4,4)=S2*S3*(-S4)-C2*C4
C
      PY(1,5)=C1*S2*C4-S1*C3*S4-C1*C2*S3*S4
      PY(1,1)=-S1*S2*C4-C1*C3*S4+S1*C2*S3*S4
      PY(1,2)=C1*C2*C4-C1*(-S2)*S3*S4
      PY(1,3)=-S1*(-S3)*S4-C1*C2*C3*S4
      PY(1,4)=C1*S2*(-S4)-S1*C3*C4-C1*C2*S3*C4
C
      PY(2,5)=S1*C2*C4+S1*S2*S3*S4
      PY(2,1)=C1*C2*C4+C1*S2*S3*S4
      PY(2,2)=S1*(-S2)*C4+S1*C2*S3*S4
      PY(2,3)=S1*S2*C3*S4
      PY(2,4)=S1*C2*(-S4)+S1*S2*S3*C4
C
      PY(3,5)=-S1*C2*C3*S4-C1*S3*S4
      PY(3,1)=-C1*C2*C3*S4+S1*S3*S4
      PY(3,2)=-S1*(-S2)*C3*S4
      PY(3,3)=-S1*C2*(-S3)*S4-C1*C3*S4
      PY(3,4)=-S1*C2*C3*C4-C1*S3*C4
C
      PY(4,5)=C1*C3*C4-S1*C2*S3*C4-S1*S2*S4
      PY(4,1)=-S1*C3*C4-C1*C2*S3*C4-C1*S2*S4
      PY(4,2)=-S1*(-S2)*S3*C4-S1*C2*S4
      PY(4,3)=C1*(-S3)*C4-S1*C2*C3*C4
      PY(4,4)=C1*C3*(-S4)-S1*C2*S3*(-S4)-S1*S2*C4
C
100   DO 200 I=1,4
      DO 201 J=1,5
      PP(I,J)=PX(I,J)*SS+PY(I,J)
201   CONTINUE
200   CONTINUE
      TSS(1)=THS1-THS01
      TSS(2)=THS2-THS02
      TSS(3)=THS3-THS03
      TSS(4)=THS4-THS04
      TSS(5)=1.0
C
C
      DO 50 I=1,4
      DO 51 J=1,5
      X(I)=XX(I,J)*TSS(J)
      Y(I)=YY(I,J)*TSS(J)
      Z(I)=ZZ(I,J)*TSS(J)
      [illegible]
      [illegible]
50    CONTINUE
```

```
      CST1=COS(THS2)*COS(THS3)+SIN(THS2)*SIN(THS3)*COS(THS4)
      CST2=SIN(THS1)*SIN(THS2)*COS(THS4)+COS(THS1)*COS(THS3)
     1     *SIN(THS4)-SIN(THS1)*COS(THS2)*SIN(THS3)
     1     *SIN(THS4)
      PS=SS*CST1+CST2
C
      DP=-PS
C
      DEN=    -X(2)*(Y(1)*Z(3)*P(4)+Y(3)*Z(4)*P(1)+Y(4)*Z(1)*P(3)
     1              -Y(4)*Z(3)*P(1)-Y(3)*Z(1)*P(4)-Y(1)*Z(4)*P(3))
     1        +X(3)*(Y(1)*Z(2)*P(4)+Y(2)*Z(4)*P(1)+Y(4)*Z(1)*P(2)
     1              -Y(4)*Z(2)*P(1)-Y(2)*Z(1)*P(4)-Y(1)*Z(4)*P(2))
     1        -X(4)*(Y(1)*Z(2)*P(3)+Y(2)*Z(3)*P(1)+Y(3)*Z(1)*P(2)
     1              -Y(3)*Z(2)*P(1)-Y(2)*Z(1)*P(3)-Y(1)*Z(3)*P(2))
C
      RNU1=    DX*(Y(2)*Z(3)*P(4)+Y(3)*Z(4)*P(2)+Y(4)*Z(2)*P(3)
     1              -Y(4)*Z(3)*P(2)-Y(3)*Z(2)*P(4)-Y(2)*Z(4)*P(3))
     1        -DY*(X(2)*Z(3)*P(4)+X(3)*Z(4)*P(2)+X(4)*Z(2)*P(3)
     1              -X(4)*Z(3)*P(2)-X(3)*Z(2)*P(4)-X(2)*Z(4)*P(3))
     1        +DZ*(X(2)*Y(3)*P(4)+X(3)*Y(4)*P(2)+X(4)*Y(2)*P(3)
     1              -X(4)*Y(3)*P(2)-X(3)*Y(2)*P(4)-X(2)*Y(4)*P(3))
     1        -DP*(X(2)*Y(3)*Z(4)+X(3)*Y(4)*Z(2)+X(4)*Y(2)*Z(3)
     1              -X(4)*Y(3)*Z(2)-X(3)*Y(2)*Z(4)-X(2)*Y(4)*Z(3))
C
C
      RNU2=   -DX*(Y(1)*Z(3)*P(4)+Y(3)*Z(4)*P(1)+Y(4)*Z(1)*P(3)
     1              -Y(4)*Z(3)*P(1)-Y(3)*Z(1)*P(4)-Y(1)*Z(4)*P(3))
     1        +DY*(X(1)*Z(3)*P(4)+X(3)*Z(4)*P(1)+X(4)*Z(1)*P(3)
     1              -X(4)*Z(3)*P(1)-X(3)*Z(1)*P(4)-X(1)*Z(4)*P(3))
     1        -DZ*(X(1)*Y(3)*P(4)+X(3)*Y(4)*P(1)+X(4)*Y(1)*P(3)
     1              -X(4)*Y(3)*P(1)-X(3)*Y(1)*P(4)-X(1)*Y(4)*P(3))
     1        +DP*(X(1)*Y(3)*Z(4)+X(3)*Y(4)*Z(1)+X(4)*Y(1)*Z(3)
     1              -X(4)*Y(3)*Z(1)-X(3)*Y(1)*Z(4)-X(1)*Y(4)*Z(3))
C
C
      RNU3=   +DX*(Y(1)*Z(2)*P(4)+Y(2)*Z(4)*P(1)+Y(4)*Z(1)*P(2)
     1              -Y(4)*Z(2)*P(1)-Y(2)*Z(1)*P(4)-Y(1)*Z(4)*P(2))
     1        -DY*(X(1)*Z(2)*P(4)+X(2)*Z(4)*P(1)+X(4)*Z(1)*P(2)
     1              -X(4)*Z(2)*P(1)-X(2)*Z(1)*P(4)-X(1)*Z(4)*P(2))
     1        +DZ*(X(1)*Y(2)*P(4)+X(2)*Y(4)*P(1)+X(4)*Y(1)*P(2)
     1              -X(4)*Y(2)*P(1)-X(2)*Y(1)*P(4)-X(1)*Y(4)*P(2))
     1        -DP*(X(1)*Y(2)*Z(4)+X(2)*Y(4)*Z(1)+X(4)*Y(1)*Z(2)
     1              -X(4)*Y(2)*Z(1)-X(2)*Y(1)*Z(4)-X(1)*Y(4)*Z(2))
C
C
      RNU4=   -DX*(Y(1)*Z(2)*P(3)+Y(2)*Z(3)*P(1)+Y(3)*Z(1)*P(2)
     1              -Y(3)*Z(2)*P(1)-Y(2)*Z(1)*P(3)-Y(1)*Z(2)*P(3))
     1        +DY*(X(1)*Z(2)*P(3)+X(2)*Z(3)*P(1)+X(3)*Z(1)*P(2)
     1              -X(3)*Z(2)*P(1)-X(2)*Z(1)*P(3)-X(1)*Z(3)*P(2))
     1        -DZ*(X(1)*Y(2)*P(3)+X(2)*Y(3)*P(1)+X(3)*Y(1)*P(2)
     1              -X(3)*Y(2)*P(1)-X(2)*Y(1)*P(3)-X(1)*Y(3)*P(2))
     1        +DP*(X(1)*Y(2)*Z(3)+X(2)*Y(3)*Z(1)+X(3)*Y(1)*Z(2)
     1              -X(3)*Y(2)*Z(1)-X(2)*Y(1)*Z(3)-X(1)*Y(3)*Z(2))
C
C
      DTH1=RNU1/DEN
      DTH2=RNU2/DEN
      DTH3=RNU3/DEN
      DTH4=RNU4/DEN
      T5=THS5
      T6=THS6
C
      THS1=THS1+DTH1
      THS2=THS2+DTH2
      THS3=THS3+DTH3
      THS4=THS4+DTH4
```

```
C
      THS5=ATAN2((-(SIN(THS1)*COS(THS2)*COS(THS4)+COS(THS1)
     1        *COS(THS2)*SIN(THS3)*COS(THS4)
     1        +COS(THS1)*SIN(THS2)*SIN(THS4)),
     1        (COS(THS1)*COS(THS2)*COS(THS3)-SIN(THS1)
     1        *SIN(THS3)))
C
C
      THS6=ATAN2((-COS(THS1)*SIN(THS2)*COS(THS4)
     1 .      +SIN(THS1)*COS(THS3)*SIN(THS4)+COS(THS1)
     1        *COS(THS2)*SIN(THS3)*SIN(THS4)),
     1        ((COS(THS1)*COS(THS2)*COS(THS3)-SIN(THS1)*SIN(THS3))
     1        *COS(THS5)-(SIN(THS1)*COS(THS3)*COS(THS4)
     1        +COS(THS1)*COS(THS2)*SIN(THS3)*COS(THS4)
     1        +COS(THS1)*SIN(THS2)*COS(THS4))*SIN(THS5)))
C
C
      DTH5=THS5-T5
      DTH6=THS6-T6
C
C
C
      IDTH1=ININT(DTH1*180.0/3.1416)
      IDTH2=ININT(DTH2*180.0/3.1416)
      IDTH3=ININT(DTH3*180.0/3.1416)
      IDTH4=ININT(DTH4*180.0/3.1416)
      IDTH5=ININT(DTH5*180.0/3.1416)
      IDTH6=ININT(DTH6*180.0/3.1416)
C
      TYPE *,IDTH1,IDTH2,IDTH3,IDTH4,IDTH5,IDTH6
C
      TYPE *,'MAY I RUN?'
      ACCEPT *,I
      IF(I.NE.1) GO TO 20
C
C
500   IF(NNN.NE.1) GO TO 510
      CALL HANDLI
      N=1000
      FN=1000.0
      THS1=0.0
      THS2=0.0
      THS3=0.0
      THS4=0.0
      THS5=0.0
      THS6=0.0
C
      THXHO(2)=THXHI(2)
      THXHO(3)=THXHI(3)
      THXHO(4)=THXHI(4)
      THXHO(5)=THXHI(5)
      THXHO(6)=THXHI(6)
      THXHO(7)=THXHI(7)
C
C
510   THXSO(2)=THS4
      THXSO(3)=THS5-THS6*0.325-THS3*0.??
      THXSO(4)=THS5+THS6*0.325-THS3*0.??
      THXSO(5)=THS1
      THXSO(6)=THS1+THS3
      THXSO(7)=THS2
C
      IF(NNN.EQ.1) GO TO 520
```

```
C
      THXMO(2)=THM1
      THXMO(3)=THXMI(3)
      THXMO(4)=THXMI(4)
      THXMO(5)=THM1
      THXMO(6)=THM1+THM3
      THXMO(7)=THM3
C
C
520   CONTINUE
C
      DO 251 I=2,7
      AD(I)=THXSO(I)-THXSI(I)
251   CONTINUE
C
C
      I=1
88    CONTINUE
      DO 302 J=2,7
C
      THXSO(J)=THXSI(J)+(AD(J)/RN)*FLOAT(I)
302   CONTINUE
      CALL MANGLO
      CALL AOUTSQ(4,17,IDATA)
      I=I+1
      IF(I.GT.N) GO TO 301
      GO TO 88
301   CONTINUE
      GO TO 10
      END
      SUBROUTINE MANGLI
      COMMON THXSI(7),THXMI(7),THXSO(7),THXMO(7)
      COMMON IDATA(14)
      DIMENSION COEF(7),ICONST(7)
      DATA COEF/0.00046,0.00142,0.00177,-0.00177,
     1    -0.00043,-0.00079,[illegible]/
      DATA ICONST/0,0,300,-300,0,0,[illegible]/
C
C     CONVERT ANGLE READINGS INTO RADIANS
C
      CALL AINSQ(15,29,IDATA)
      DO 10 I=1,7
      THXSI(I)=FLOAT(IDATA(I)[illegible]-ICONST(I))*COEF(I)
      THXMI(I)=FLOAT(-IDATA(I+7)[illegible]-ICONST(I))*COEF(I)
10    CONTINUE
      RETURN
      ENTRY MANGLO
C     CONVERT ANGLE RADIANS INTO OUTPUT FORMS
      DO 20 I=1,7
      IDATA(I)=(ININT(THXSO(I)/COEF(I))+ICONST(I))[illegible]
      IDATA(I+7)=-(ININT(THXMO(I)/COEF(I))+ICONST(I))[illegible]
20    CONTINUE
C
      DO 77 J=1,14
      IF(IDATA(J).GT.[illegible]) IDATA(J)=[illegible]
77    CONTINUE
C
      RETURN
      END
```

```
C
      COMMON ...
      COMMON ...
C
      CVD=10.0/32752.0
C
      DA = 15.125
      DB = 14.750
      DC =  4.00
      DXT=- 1.750
      DYT=-19.3125
      DZT= 10.750
      DX0= 13.0625
      DY0=  3.4375
C
C***      PHAI DEG
      DM(1)= ATAN((DC-DY0)/DX0)*57.2958
C***      DF
      DM(2)= ...
C***      DCAN
      DM(3)=SQRT((DC-DY0)**2+DX0**2)
C***
      DM(4)= ...
C***
      DM(5)=57.2958*ASIN(DM(4))
C***
      DM(6)= ...
C***
      DM(7)=DM(6)/57.2958
C
C
C***      DF
      DM(8)=...
C***
      DM(9)=...
C***
      DM(10)=57.2958*ACOS(DM(9))
C***      THA DEG
      DM(11)=DM(...)-DM(...)
C
C
      ...=DM(...)
      ...=DM(...)
      ...
C
      TYPE *, ...
C
      ...
      ...
      ...
C
      TYPE *, ...
C
      CALL ANINIT
C
      TYPE *, ...
      ...
      IF ...
C
      CALL ...
C
      DO ...
      ...
C
      CONTINUE
```

```
C
      AA=VI(1)
      BB=VI(2)
      VI(1)=BB
      VI(2)=AA
      AA=VI(3)
      BB=VI(4)
      VI(3)=BB
      VI(4)=AA
      AA=VI(5)
      BB=VI(6)
      VI(5)=BB
      VI(6)=AA
C
      DO 180 I=1,6
      IVI(I)=IFINT(VI(I))
180   CONTINUE
      TYPE *,IVI(1),IVI(2),IVI(3),IVI(4),
     1        IVI(5),IVI(6)
C
C     THETA C IN DEG
      C1=-20.0*VI(1)
      C2=-20.0*VI(2)-CC2
      C3= 20.0*VI(3)+90.0
      C4=-20.0*VI(4)-CC4
      C5= 20.0*VI(5)-CC5
      C6=-20.0*VI(6)
C
200   TYPE *,'MEASURE'
      ACCEPT *,I
      IF(I.NE.1) GO TO 900
      CALL AIMES(0,5,IDT)
C
      DO 250 I=1,6
      VI(I)=CONVFLOAT(IDT(I))
250   CONTINUE
C
      AA=VI(1)
      BB=VI(2)
      VI(1)=BB
      VI(2)=AA
      AA=VI(3)
      BB=VI(4)
      VI(3)=BB
      VI(4)=AA
      AA=VI(5)
      BB=VI(6)
      VI(5)=BB
      VI(6)=AA
C
      THD(1)=-20.0*VI(1)-C1
      THD(2)=-20.0*VI(2)-C2
      THD(3)= 20.0*VI(3)-C3
      THD(4)=-20.0*VI(4)-C4
      THD(5)= 20.0*VI(5)-C5
      THD(6)=-20.0*VI(6)-C6
C
      DO 255 I=1,6
      ITHD(I)=IFINT(THD(I))
255   CONTINUE
      TYPE *,'ITHD(1)_ITHD(6)'
      TYPE *,ITHD(1),ITHD(2),ITHD(3),ITHD(4),
     1        ITHD(5),ITHD(6)
C
C
      DO 260 I=1,6
      THR(I)=(3.14159/180.0)*THD(I)
```

```
C
      IF(THR(1).EQ.0.0) GO TO 270
C
      S(1)=SQRT((TAN(THR(1)))**2+(TAN(THR(2)))**2)
C     TH1-D=S(2)
      S(2)=ATAN(1.0/S(1))
C
      S(3)=(TAN(THR(2)))/SIN(THR(1))
C
C     TH2-D=S(4)
      S(4)=-ATAN(S(3))
      GO TO 290
 270  S(2)=(3.14159/2.0)-THR(2)
      IF(THR(2).GT.0.0) S(4)= 3.14159/2.0
      IF(THR(2).LT.0.0) S(4)=-3.14159/2.0
      IF(THR(2).EQ.0.0) GO TO 900
 290  CONTINUE
C     L=S(5)
      S(5)=DA*COS(S(2))+DB*SIN(S(2)+THR(4))
C
C
      X=DXT+S(5)*COS(S(4))
      Y=DYT+S(5)*SIN(S(4))
      Z=DZT-DA*SIN(S(2))
     1          -(DB*COS(S(2)+THR(4))+DC)
C
      A=(3.14159/2.0)-(S(4)-THR(6))
C
      TYPE *,'X,Y,Z,S'
      TYPE *,X,Y,Z,S
C
      X=X*10.0
      Y=Y*10.0
      Z=Z*10.0
C
      IX=ININT(X)
      IY=ININT(Y)
      IZ=ININT(Z)
      IA=ININT(A*180.0/3.14159)
C
      TYPE *,IX,IY,IZ,IA
C

      GO TO 100
 900  CONTINUE
      STOP
      END
```

```
      PROGRAM TTT
C
      COMMON  /N1/ THXSI(7),THYMI(7),THXSD(7),THYMD(7)
      COMMON  /N2/ THS(6),XS,YS,ZS,SS,XSD,YSD,ZSD,SSD
      COMMON  /N3/ THM(6),XM,YM,ZM,SM,XMD,YMD,ZMD,SMD
      COMMON  /N4/ IDATA(14)
      COMMON  /N5/ IN,XHE,YHE,ZHE,AHE,IVX,IVY,IVZ,IT,IN
      COMMON  /N6/ XMTT,YMTT,ZMTT,IR
C
      DIMENSION       AD(7)
C
      CALL    ANINIT
C
      CALL DOUT(24,0)
      CALL DOUT(26,0)
      TYPE *,'COMPUTER CONTROL ?'
      ACCEPT *,I
      IF(I.NE.1) GO TO 900
      CALL [illegible](16,29,IDATA)
      CALL ADUTSQ(4,17,IDATA)
C
      TYPE *,'[illegible].OF.DIV;SPEED'
      ACCEPT [illegible]
      RN=FLOAT(N)
C
      TYPE *,'NEED ORIGIN SET ?'
      ACCEPT *,I
      IF(I.EQ.1) GO TO 500
C
      TYPE *,'START ?'
      ACCEPT *,I
      IF(I.NE.1) GO TO 900
C
      CALL    SUBI
C
      THS(1)=THXSI(5)
      THS(2)=THXSI(7)
      THS(3)=THXSI(6)-THS(1)
      THS(4)=THXSI(2)
      THS(5)=(THXSI(4)+THXSI(3))/2.0+0.27*THS(3)
      THS(6)=(THXSI(4)-THXSI(3))/1.65
C
      THM(1)=THYMI(5)
      THM(2)=THYMI(7)
      THM(3)=THYMI(6)-THM(1)
      THM(4)=THYMI(2)
      THM(5)=(THYMI(4)+THYMI(3))/2.0-0.27*THM(3)
      THM(6)=(THYMI(4)-THYMI(3))/1.65
C
      TYPE *,'PRESENT THS(I)'
C
      ITHS1=ININT(57.296*THS(1))
      ITHS2=ININT(57.296*THS(2))
      ITHS3=ININT(57.296*THS(3))
      ITHS4=ININT(57.296*THS(4))
      ITHS5=ININT(57.296*THS(5))
      ITHS6=ININT(57.296*THS(6))
C
      TYPE *,ITHS1,ITHS2,ITHS3,ITHS4,ITHS5,ITHS6
C
      CALL    SUBS1
C
      TYPE *,'[illegible] DX,DY,DZ,DS ?'
      ACCEPT *,I
      IF(I.NE.1) GO TO 900
```

```
      TYPE *,'DX,DY,DZ,DS'
      ACCEPT *,DX,DY,DZ,DS
C
      XSD=XS-DX
      YSD=YS+DY
      ZSD=ZS+DZ
      SSD=SS+DS
C
      IF(I.EQ.1) GO TO 250
C
      TYPE *,'PRESENT XMTT,YMTT,ZMTT'
      TYPE *,XMTT,YMTT,ZMTT
C
      TYPE *,'NEED NEW XMTT,YMTT,ZMTT ?'
      ACCEPT *,I
      IF(I.NE.1) GO TO 200
C
      TYPE *,'NEW XMTT,YMTT,ZMTT'
      ACCEPT *,XMTT,YMTT,ZMTT
C
200   XS=XS+XMTT
      YS=YS+YMTT
      ZS=ZS+ZMTT
C
      IXS=ININT(XS)
      IYS=ININT(YS)
      IZS=ININT(ZS)
C
      TYPE *,'IXS,IYS,IZS,FROM SA'
      TYPE *,IXS,IYS,IZS
C
180   TYPE *,'NEED INITIAL SET FOR SUBME ?'
      ACCEPT *,I
      IF(I.NE.1) GO TO 220
      IM=1
      CALL SUBME
      GO TO 18[illegible]
220   IM=2
      CALL SUBME
C
      IXME=ININT(XME)
      IYME=ININT(YME)
      IZME=ININT(ZME)
C
      TYPE *,'IXME,IYME,IZME,FROM ME'
      TYPE *,IXME,IYME,IZME
C
      TYPE *,'MOVE ?'
      ACCEPT *,I
      IF(I.NE.1) GO TO 900
C
      XSD=XME
      YSD=YME
      ZSD=ZME
      SSD=SS
C
250   TL1=THS(1)
      TL2=THS(2)
      TL3=THS(3)
      TL4=THS(4)
      TL5=THS(5)
      TL6=THS(6)
```

```
      CALL    SUBB2
C
      IT1=ININT((THS(1)-TL1)*57.296)
      IT2=ININT((THS(2)-TL2)*57.296)
      IT3=ININT((THS(3)-TL3)*57.296)
      IT4=ININT((THS(4)-TL4)*57.296)
      IT5=ININT((THS(5)-TL5)*57.296)
      IT6=ININT((THS(6)-TL6)*57.296)
C
      TYPE *,IT1,IT2,IT3,IT4,IT5,IT6
C
      GO TO 600
C
500   THS(1)=0.0
      THS(2)=0.0
      THS(3)=0.0
      THS(4)=0.0
      THS(5)=0.0
      THS(6)=0.0
600   TYPE *,'MAY I RUN ?'
      ACCEPT *,I
      IF(I.NE.1) GO TO 900
C
      THXSO(1)=0.0
      THXMO(1)=0.0
C
      DO 300 J=2,7
      THXMO(J)=THXMI(J)
300   CONTINUE
C
      THXSO(2)=THS(4)
      THXSO(3)=THS(5)-THS(6)*0.305-THS(3)*0.27
      THXSO(4)=THS(5)+THS(6)*0.305-THS(3)*0.27
      THXSO(5)=THS(1)
      THXSO(6)=THS(1)+THS(3)
      THXSO(7)=THS(2)
C
      TYPE *,'POINT 1'
C
      DO 251 I=2,7
      AD(I)=THXSO(I)-THXSI(I)
251   CONTINUE
      I=1
C
      TYPE *,'POINT 2'
C
38    CONTINUE
      DO 302 J=2,7
      THXSO(J)=THXSI(J)+(AD(J)/FN)*FLOAT(I)
302   CONTINUE
C
      CALL    SUBO
C
      CALL    ACUTSO(4,17,IDATA)
      I=I+1
      IF(I.GT.N)  GO TO 301
      GO TO 38
C
      TYPE *,'POINT 3'
C
301   CONTINUE
C
      TYPE *,'POINT 4'
C
900   CONTINUE
C
      TYPE ...
      
      END
```

Copy available to [redacted]
[illegible] fully legible reproduction

```
      PROGRAM HICOM
C
      COMMON  THXSI(7),THXMI(7),THYSO(7),THXMO(7)
      COMMON  THS(6),XS,YS,ZS,SS,XSO,YSO,ZSO,SSO
      COMMON  THM(6),XM,YM,ZM,SM,XMO,YMO,ZMO,SMO
      COMMON  IDATA(14)
      COMMON  IM,XME,YME,ZME,AME,IVX,IVY,IVZ,IT,IN
      COMMON  XMTT,YMTT,ZMTT,IP
C
C
      THXSO(1)=0.0
      THXMO(1)=0.0
C
      CALL ANINIT
C
      TYPE *,'KEEP PRESENT POSITION ?'
      ACCEPT *,I
      IF(I.NE.1) GO TO 900
C
      CALL AINSQ(16,29,IDATA)
      CALL AOUTSQ(4,17,IDATA)

C
      TYPE *,'NEED TO SET A & B ?'
      ACCEPT *,I
      IF(I.NE.1) GO TO 140
C
      TYPE *,'A(=0.7),B(=0.5)'
      ACCEPT *,A,B
      A1=1.0-A
C
140   TYPE *,'MEA INITIAL SET'
      ACCEPT *,I
      IF(I.NE.1) GO TO 130
C
      IM=1
      CALL SUBME
C
130   TYPE *,'MODE WITHOUT TABLE'
      ACCEPT *,I
      IF(I.NE.1) GO TO 900
C
C     MODE WITHOUT TABLE
C
90    T1=SECNDS(0.0)
100   T3=SECNDS(0.0)
C
      CALL SUBI
C
      THS(1)=THXSI(5)
      THS(2)=THXSI(7)
      THS(3)=THXSI(6)-THS(1)
      THS(4)=THXSI(2)
C
      CALL SUBS1
C
      THM(1)=THXMI(5)
      THM(2)=THXMI(7)
      THM(3)=THXMI(6)-THM(1)
      THM(4)=THXMI(2)
C
      CALL SUBM1
C
C
      XSD=A*XM+A1*XS
      YSD=A*YM+A1*YS
      ...
```

Copy available to DTIC does not permit fully legible reproduction

```
      XMD=A*XS-A1*XM
      YMD=A*YS+A1*YM
      ZMD=A*ZS+A1*ZM
      SMD=A*SS+A1*SM
C
C
      YSD=XSD+B*(XS-XS2)
      YSD=YSD+B*(YS-YS2)
      ZSD=ZSD+B*(ZS-ZS2)
      SSD=SSD+B*(SS-SS2)
C
      XMD=XMD+B*(XM-XM2)
      YMD=YMD+B*(YM-YM2)
      ZMD=ZMD+B*(ZM-ZM2)
      SMD=SMD+B*(SM-SM2)
C
C     CALL SUBS2
      CALL SUBM2
C
      THSI6=(THXSI(4)-THXSI(3))/1.65-THS(6)
      THMI6=(THXMI(4)-THXMI(3))/1.65-THM(6)
      THSO6=A*THMI6+A1*THSI6+THS(6)
      THMO6=A*THSI6+A1*THMI6+THM(6)
C
      THSO6=THSO6+B*(THSI6-THSI62)
      THMO6=THMO6+B*(THMI6-THMI62)
C
C
      THXSO(2)=THS(4)
      THXSO(3)=THS(5)-THSO6*0.825-THS(3)*0.27
      THXSO(4)=THS(5)-THSO6*0.825-THS(3)*0.27
      THXSO(5)=THS(1)
      THXSO(6)=THS(1)+THS(3)
      THXSO(7)=THS(2)
C
      THXMO(2)=THM(4)
      THXMO(3)=THM(5)-THMO6*0.825-THM(3)*0.27
      THXMO(4)=THM(5)+THMO6*0.825-THM(3)*0.27
      THXMO(5)=THM(1)
      THXMO(6)=THM(1)+THM(3)
      THXMO(7)=THM(2)
C
C
      CALL SUSO
      CALL AOUTSQ(4,17,IDATA)
C
      XS2=XS1
      YS2=YS1
      ZS2=ZS1
      SS2=SS1
C
      XS1=XS
      YS1=YS
      ZS1=ZS
      SS1=SS
C
      XM2=XM1
      YM2=YM1
      ZM2=ZM1
      SM2=SM1
C
      XM1=XM
      YM1=YM
      ZM1=ZM
      SM1=SM
```

```
C
        THSI62=THSI61
        THMI62=THMI62
C
        THSI61=THSI6
        THMI61=THMI6
C
C
170     T4=SECNDS(T3)
        IT4=AINT(T4*20.0)
        IF(IT4.LT.1) GO TO 100
C
        T2=SECNDS(T1)
        IT2=ININT(T2)
C
        IF(IT2.LT.60) GO TO 100
C
        TYPE *,'MODE WITHOUT TABLE OR WITH TABLE'
        ACCEPT *,I
        IF(I.EQ.1) GO TO 90
C
        TYPE *,'MODE WITH TABLE'
C       MODE WITH TABLE
C
        TYPE *,'COINCIDENCE OF TWO ORIGINS ?'
        ACCEPT *,I
        IF(I.NE.1) GO TO 180
C
        TYPE *,'XMTT<0.0, YMTT<0.0, ZMTT<0.0'
        ACCEPT *,XMTT,YMTT,ZMTT
C
180     CONTINUE
C
        TYPE *,'XS,YS,ZS,SS'
        IXS=ININT(XS)
        IYS=ININT(YS)
        IZS=ININT(ZS)
        ISS=ININT(SS)
        TYPE *,IXS,IYS,IZS,ISS
C
        TYPE *,'XM,YM,ZM,SM'
        IXM=ININT(XM)
        IYM=ININT(YM)
        IZM=ININT(ZM)
        ISM=ININT(SM)
        TYPE *,IXM,IYM,IZM,ISM
C
        TYPE *,'XME,YME,ZME,AME'
        IXME=ININT(XME)
        IYME=ININT(YME)
        IZME=ININT(ZME)
        IAME=ININT(AME)
        TYPE *,IXME,IYME,IZME,IAME
C
205     TYPE *,'TABLE SPEED ?'
        ACCEPT *,I
        IF(I.NE.1) GO TO 210
C
        TYPE *,'IVX,IVY,IVZ,IT,IN'
        ACCEPT *,IVX,IVY,IVZ,IT,IN
C
210     TYPE *,'TABLE GO'
        ACCEPT *,I
        IF(I.NE.1) GO TO 205
C
```

```
        INM=IN
        IN=0
C
        CALL MOVE(IVX,IVY,IVZ)
C
        X1=0.0
        Y1=0.0
        Z1=0.0
C
        X2=0.0
        Y2=0.0
        Z2=0.0
C
        III=0
310     T6=SECNDS(0.0)
C
        CALL MOVE(IVX,IVY,IVZ)
320     T5=SECNDS(0.0)
C
        CALL SUBI
C
        THS(1)=TH*SI(5)
        THS(2)=TH*SI(7)
        THS(3)=TH*SI(6)-THS(1)
        THS(4)=TH*SI(2)
C
        CALL SUBS1
C
        THM(1)=TH*MI(5)
        THM(2)=TH*MI(7)
        THM(3)=TH*MI(6)-THM(1)
        THM(4)=TH*MI(2)
C
        CALL SUBM1
C
        XSE=XS-X2
        YSE=YS-Y2
        ZSE=ZS-Z2
C
        IM=2
C
        CALL SUBME
C
C****** IVX----REAL SPEED
C******
C
        XSD=A*XM+A1*XSE+X
        YSD=A*YM+A1*YSE+Y
        ZSD=A*ZM+A1*ZSE+Z
        SSD=A*SM+A1*SS
C
        XMD=A*XSE+A1*XM
        YMD=A*YSE+A1*YM
        ZMD=A*ZSE+A1*ZM
        SMD=A*SS+A1*SM
C
        III=III+1
C
        IF(III.GT.0) GO TO 710
C
        XSE2=XSE
        YSE2=YSE
        ZSE2=ZSE
        SS2=SS
C
        XM2=XM
[illegible]
[illegible]
[illegible]
```

Copy available to DTIC does not permit fully legible reproduction

```
C
      XSE1=XSE
      YSE1=YSE
      ZSE1=ZSE
      SS1=SS
C
      XM1=XM
      YM1=YM
      ZM1=ZM
      SM1=SM
C
710   CONTINUE
C
      XSD=XSD+R*(XSE-[illegible]
      YSD=YSD+R*(YSE-[illegible]
      ZSD=ZSD+R*(ZSE-ZSE[illegible]
      SSD=SSD+R*(SS-SS[illegible]
C
      XMD=XMD+R*(XM-XM[illegible]
      YMD=YMD+R*(YM-YM[illegible]
      ZMD=ZMD+R*(ZM-SMD)
C
      CALL SUB[illegible]
      CALL SUB[illegible]
C
      THSI5=(THXSI(4)[illegible]
      THMI5=(THXMI(4)[illegible]
      THSC5=A*THMI5+A[illegible]
      THMC5=A*THSI5-A[illegible]
C
      THSO6=THSC5-R*[illegible]
      THMO6=THMC5+R*[illegible]
C
      THXSO(2)=THS(4)
      THXSO(3)=THS[illegible]
      THXSO(4)=THS[illegible]
      THXSO(5)=THS[illegible]
      THXSO(6)=THS[illegible]
      THXSO(7)=THS[illegible]
C
      THXMO(2)=THM(4)
      THXMO(3)=THM[illegible]
      THXMO(4)=THM[illegible]
      THXMO(5)=THM[illegible]
      THXMO(6)=THM[illegible]
      THXMO(7)=THM[illegible]
C
      CALL SUB[illegible]
      CALL ACC[illegible]
C
      X2=X[illegible]
      [illegible]
      [illegible]
      [illegible]
      Y[illegible]
      [illegible]
      Z[illegible]
C
      XSE[illegible]
      YSE[illegible]
      ZSE[illegible]
      [illegible]
      SS[illegible]
      THS[illegible]
```

```
[illegible]
```

Copy available to DTIC does not permit fully legible reproduction

END

7-87

DTIC